DOCUMENT RESUME

ED 337 160 IR 015 214

AUTHOR	Gayle, Susan, Ed.
TITLE	NECC '91. Proceedings of the Annual National Educational Computing Conference (12th, Phoenix, Arizona, June 16-20, 1991).
SPONS AGENCY	National Educational Computing Conference.
REPORT NO	ISBN-0-924667-69-9
PUB DATE	91
NOTE	372p.; For the 1989 proceedings, see ED 317 190. Proceedings from 1984 to 1990 are also available from the International Society for Technology in Education for $25.00 each.
AVAILABLE FROM	International Society for Technology in Education, 1787 Agate Street, Eugene, OR 97403 ($25.00).
PUB TYPE	Collected Works - Conference Proceedings (021)
EDRS PRICE	MF01/PC15 Plus Postage.
DESCRIPTORS	*Computer Assisted Instruction; Computer Literacy; Computer Software; *Educational Technology; Elementary Secondary Education; Higher Education; *Hypermedia; Interactive Video; Language Arts; Mathematics Instruction; Microcomputers; Multicultural Education; *Multimedia Instruction; Needs Assessment; Problem Solving; Programing; School Restructuring; Science Instruction; Social Studies; Telecommunications; Writing Instruction
IDENTIFIERS	Logo Programing Language

ABSTRACT

The theme of this annual conference was "Solutions," and referred to both human and electronic solutions. The proceedings are organized by conference day and contain the following: (1) Tuesday--presentations on multicultural and multimedia education, telecommunications, hypermedia, problem solving, mathematics instruction, language arts, social studies, teacher training, computer literacy, interactive video, Logo, school restructuring, and computer science (80 abstracts, 12 papers); (2) Wednesday--presentations on mathematics instruction, computer literacy, instructional research, teacher training, educational technology, special populations, tutorial systems, computer software, hypermedia, multimedia, science instruction, Logo projects, writing instruction, higher education, sex differences, foreign exchange students, international computing, video communications, music and the arts, bilingual computing, and funding (94 abstracts, 13 papers); and (3) Thursday--presentations on multimedia instruction, problem solving, mathematics instruction, teacher training, deaf education, elementary and secondary schools, corporate environments, diagnostic skills, telecommunications, world and local communities, Logo technology, Apple II applications, multimedia faculty development, special learners, needs assessment, simulation, databases, and laser discs (78 abstracts, 9 papers). A list of conference committees, society descriptions, and reviewer acknowledgements are also included. An index by authors concludes the report. (DB)

ED337160

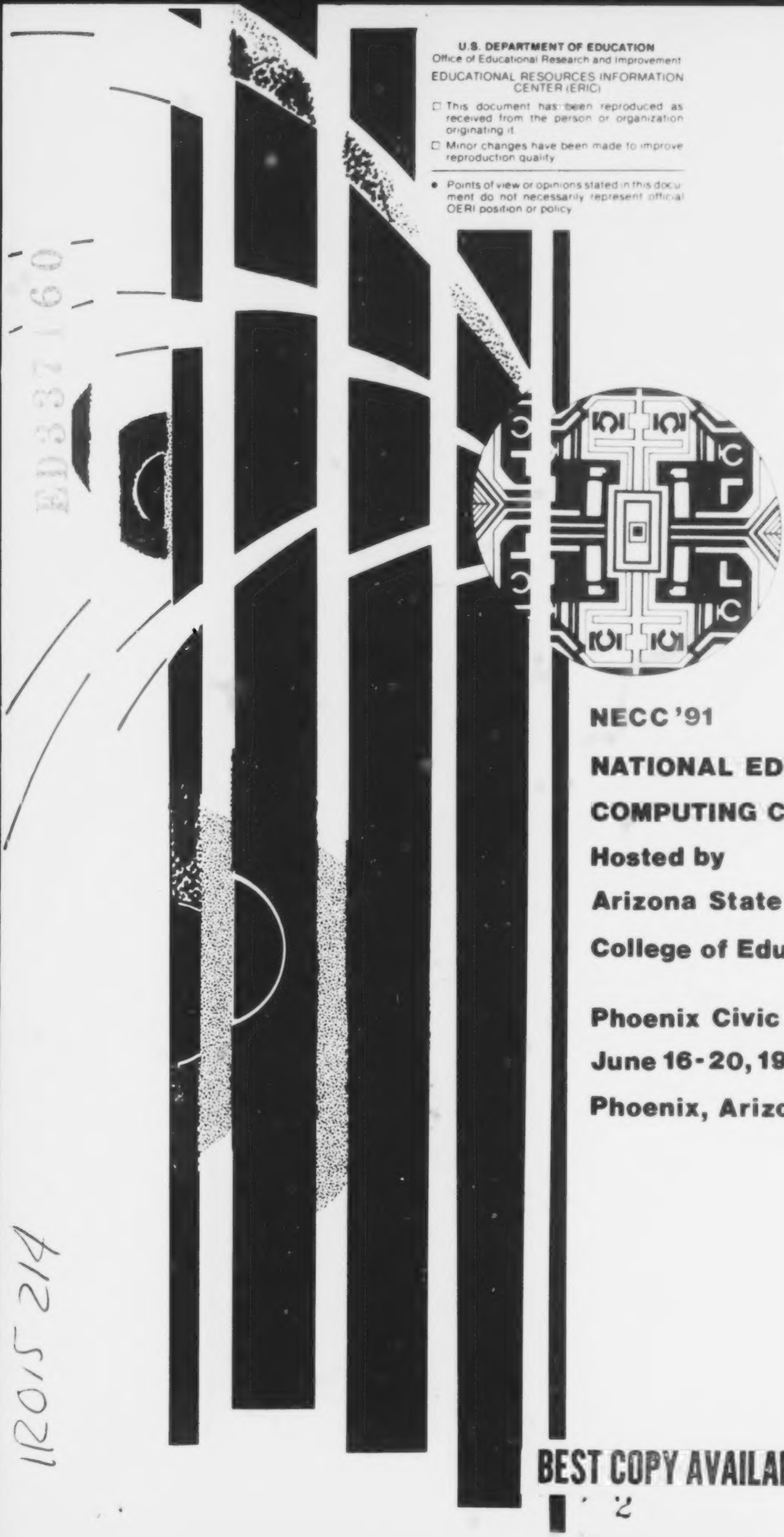
IR015214

U.S. DEPARTMENT OF EDUCATION
Office of Educational Research and Improvement
EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)

☐ This document has been reproduced as received from the person or organization originating it.

☐ Minor changes have been made to improve reproduction quality.

- Points of view or opinions stated in this document do not necessarily represent official OERI position or policy.



Proceedings

NECC '91

NATIONAL EDUCATIONAL COMPUTING CONFERENCE

Hosted by
Arizona State University
College of Education

Phoenix Civic Plaza
June 16-20, 1991
Phoenix, Arizona

BEST COPY AVAILABLE

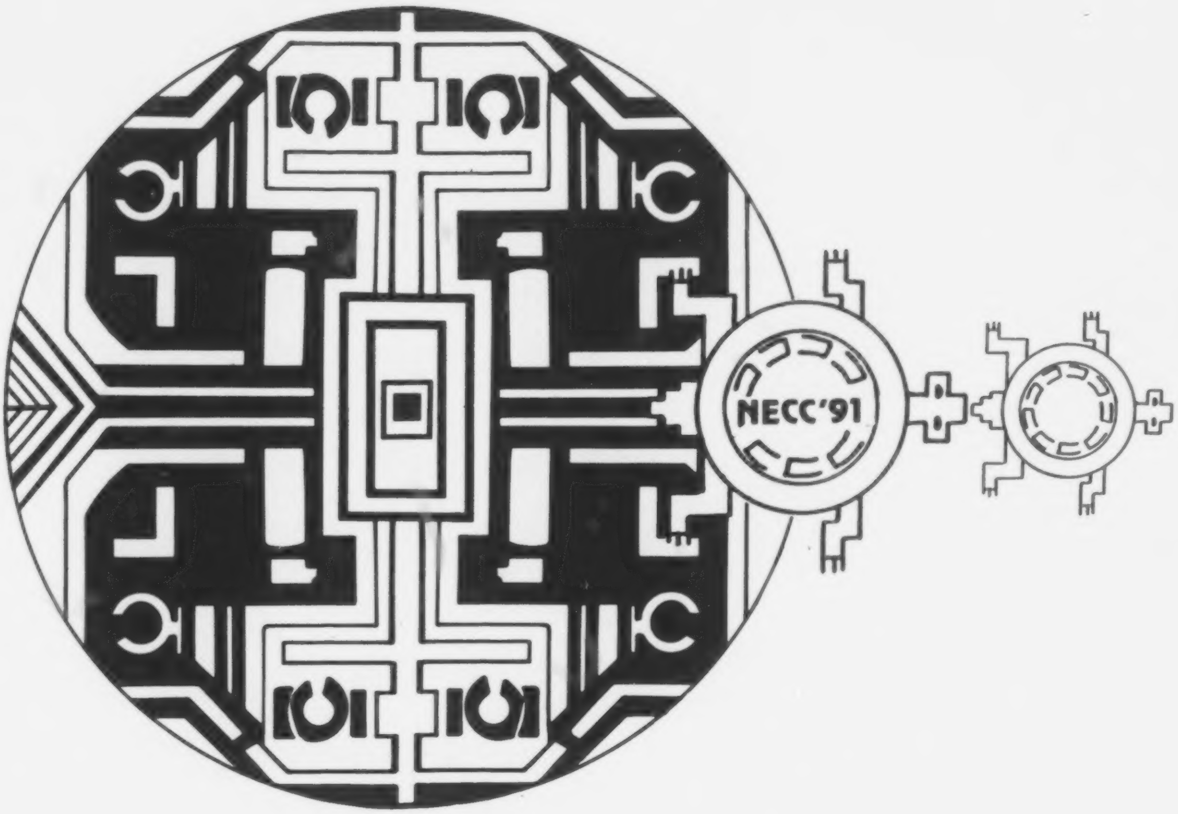
"PERMISSION TO REPRODUCE THIS MATERIAL HAS BEEN GRANTED BY

Susan Gayle

TO THE EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)."

PROCEEDINGS

Susan Gayle, Editor



PROCEEDINGS OF NECC '91

ISBN 0-924667-69-9

© 1991, NECC

Published by
International Society for Technology in Education
1787 Agate Street
Eugene, OR 97403

For
National Educational Computing Conference
June 1991

ISTE Production Team
Ian Byington
Susan Gayle
Pamela Henwood
Adrienne Kelley
Tamara Kidd
Tracye May
Ron Renchler

Foreword

The twelfth annual National Educational Computing Conference provides an opportunity to experience the newest and best in the field of educational technology for all learning environments. NECC has built a tradition of sharing ideas, techniques, and research findings among educators and administrators in all aspects of training and education.

Solutions, the theme of the twelfth annual NECC, refer to both human and electronic solutions. This conference seeks to develop solutions within curriculum areas and for business/industry training needs. It explores, in particular, multimedia in response to ever-changing challenges in education. Multimedia offers great possibilities for improving learning at all levels. These *Proceedings* provide written reports on these ideas.

A conference of this size, complexity, and emphasis on technology has been an undertaking most of us had not imagined. An enormous **thank you** goes to the people who dedicated a year or more of their free time making this really happen. The core committee had the services of very dedicated committees whose timely completion of tasks made this all happen by June 16. Serving on the program committee were Norman Higgins for university projects, Barb Coady and Jane Davis for K-12 projects, Maurene Gerson for workshops, Jane Muto for society sessions and birds-of-a-feather sessions, and Mike Charles for presiders. On the local arrangements committee, Clifford Hurston and Gloria N. Wilson for equipment, and Warren Gamas and Wayne Hartschuh for social events deserve special acknowledgement. Registration was ably handled by Jennifer Mayhew, Barbara Menzel, Elaine Phares, and Michelle Bailey. Martha Rader, our efficient treasurer, was assisted by Tim Quick. A special thank you to Jennifer Mayhew, our conference secretary, for attention to all the "details."

We would like to extend a special thank you to Georgia Graham, our graphic artist, who created exceptional promotional materials for NECC '91.

The NECC Steering Committee, with Jim Poirot as chair and Diane Harris as secretary, played an invaluable consulting role, guiding us in keeping with NECC tradition. The ISTE office with Anita Best as representative to the program committee and Susan Gayle as conference coordinator were invaluable as proofreaders of the documents, facilitators at each step along the way, and managers of the general headquarters site.

As has been the case for the last several years, Sharon Finke and Paul Katz organized the exhibitors' show and presentations in their usual competent manner.

We would like to express our gratitude to the administration, faculty, graduate students, and staff at Arizona State University, who helped us whenever we asked. Special thanks to President Lattie F. Coor, Dean L. D. Webb, and the faculty of Educational Media and Computers.

Thanks to the many presenters and presiders who shared their knowledge and expertise and who made the program so diversified and outstanding.

Our special gratitude to ISTE for all its support in whatever aspects were needed. The conference would not be possible without their guidance and services.

Finally, a big thank you to all the attendees who came to Phoenix (in spite of the concern about 122 degree temperatures). Their enthusiasm for the program has made this an enriching experience.

Gary G. Bitter, Mary M. Hatfield—*Conference Co-Chairs*
Helen L. Padgett—*Program Chair, Papers*
Janie Wilson—*Local Arrangements Chair*

Arizona State University
Community Services Center
Tempe, AZ 85287-0908

Table of Contents

Foreward .... *I*

Table of Contents—Tuesday .... *V*

Table of Contents—Wednesday .... *IX*

Table of Contents—Thursday .... *XIII*

NECC '91 Conference Committees .... *XVII*

Society Descriptions .... *XIX*

Reviewer Acknowledgements .... *XXIII*

NECC '91 Conference Proceddings—Tuesday *1*

NECC '91 Conference Proceddings—Wednesday *121*

NECC '91 Conference Proceddings—Thursday *255*

Author Index *347*

Table of Contents—Tuesday

Special Presentation
Interactive Videodisc Technology in Science Instruction: Matching Methods and Materials *1*
Thomas E. Boudrot

Projects
Math Applications in K-12 Environments 2

Can Strategy-Based Computer Instruction Enhance Early Mathematics Understanding?
Douglas Frye, Jodie Ambrosino, and Cheng Dong Lu

Multi-sensory Manipulatives and Developmental Mathematics
David Fowler

Producing Aviation Videodisc Lessons for Kindergarten Math in a Bilingual Environment
Nancy Nelson Knupfer

Special Presentation
Electronic Studying: Computer-Based Outliners Promote Academic Success Across the Curriculum *4*
Lynne Anderson-Inman and Mark Horney

Projects
Projects in the Secondary Setting *5*

Computer Magnet School: A High School Model
Pat Hart

Changing the Dominate Paradigm—Project 2001
Cathy Cordes

New Mexico Education Technology Institute
Kurt Steinhaus

Paper
A BASIC Programming Curriculum for Enhancing Problem-Solving Ability *7*
Richard C. Overbaugh

Paper
The Need for Education and Technology: Examples from the GPCeditor *16*
Mark Guzdial

Projects
Language Arts Projects: K-University *24*

The Computer as a Writing Classroom
Lance Coon

Cooperative Learning Strategies with Logo Writer
Sue Horn, Kaye Coats, and June Ehinger

Researcher & Programmer: A Collaborative Approach to Developing Research-Based Software
Janet Graeber and Roger Knight

Projects
Teacher Education and Needs Research *25*

Needs Assessment: Level of Experience with Multimedia Computing Prerequisites
Loretta Cardinale

The Computing Experiences of Teacher Education Students: A Longitudinal Study
Alison Elliot

Teaching Computer Educators: A Case for Logo
Sharon Yoder

Projects
Multicultural Multimedia *26*

Development of a Socio-Cultural HyperCard Stack for Criticism of Software
Barbara A. Boyer

Diverse-ALOT: A Multi-cultural, Multi-media Database
Darlene Nowak

The Conception and Design of Afrocentric Multimedia
Bruce Lincoln

Projects
Telecommunications and Science *28*

Hypermedia Interactive Kaleidoscope for Educators in Science
Manorama Talaiver

Telecomputing Support for Distance Learning
Inabeth Miller

Utilization of Computer Software for Athletic Training Competency Testing
Joanne M. Dunnock

Special Presentation—SIGCUE
Legalities Notwithstanding, What's Fair in Software Use *29*

Projects
Language Applications Including Multimedia/Hypermedia *30*

The Proof is in the Pudding
Shelley Wepner

Developmentally Sequenced "Thinking Webs" for K-8 Hypermedia/Multimedia Materials Design
Judi Harris

Using LogoWriter as a Tool for Teaching Problem Solving
David Kressen

Projects
Social Studies Applications in the K-12 Environments *31*

Explore, Discover, Present!
Technology in the Social Studies Classroom
Peggy Healy Stearns

Integration of National Geographic Kids Network
Across the Curriculum
Susan Short

Contemporary Issues in Social Studies Utilizing
the Latest Technology
Deidre Foreman

Special Presentation
Multimedia in the Classroom Today *32*
Roger Wagner

Special Presentation—SCS
The Algebra Workbench: A Tool for Inquiry
Mathematics in Pre-Algebra and Algebra *32*

Projects
Learning Styles and Multimedia *33*

The Hypertext Vygotsky Project
H. Eric Branscomb

Teaching HyperCard to Children
Maria Swayne and Karen Hare

Image Processing for Teaching
Richard Greenberg, Robert Strom, Steven Croft,
and Lyn Doose

Paper
Desktop Publishing: A Course Reader
(Desktop publishing or desktop plagiarizing?) *34*
Paul Matray

Paper
Computers, Writing, and Whole Language:
Staff Development Issues *40*
Cathy L. Gunn

Projects
Teacher Training *48*

Teacher Explorer Centers:
Introducing Michigan Teachers to MultiMedia
Anne Porter and Fred D'Ignazio

Implementing a Staff Development Program
in a Large School District
Sharon Bell

Total Integration of Computers/Multimedia Technology
into a K-12 Curriculum
Tony Kietrys and Jack Kmetz

Paper
The Development of an Evaluation Handbook
For Computer Integrated Instruction *49*
Richard Mowe

Paper
School Administrators and the Management
of Technological Change *54*
Kieth C. Wright

Special Presentation
Earth Lab: Application of LAN Technology
to School Restructuring *61*
Denis Newman and Paul Reese

Special Presentation—ISTE
Hypermedia and Multimedia—Current Trends in Education *61*

Special Presentation
MediaText: The Brownie Instamatic of Multimedia *62*
Elliot Soloway

Special Presentation
HyperCard/HyperStudio Based Curriculum
for Bilingual Instruction *62*
Michael Reed

Projects
Algebra, Calculus, and Statistics *63*

Math for the 90s: Teletraining Project
Catherine F. Conway

TapeMeasure: Video as a Source of Data
in Student-Designed Investigations
Andee Rubin

Algebra Processors:
An Intellectual Companion for Calculus Students
Peter Flanagan-Hyde

Special Presentation
Computer Science Curriculum Using the Logo Language *65*
Ron Place, Chuck Auck and Mike Sansing

Paper
Beyond Computer Literacy for School Administrators *66*
Charles F. Adams

Paper
Competencies Necessary For An Integrated
Microcomputer Application Course *69*
Dr. Joseph C. Otto

Projects
Computer Education Leadership and Graphic Art *74*

Animation and Interactive Video in Multi Media
Health Science Instruction
James Wall and John Strange

Guidelines for Better Screen Design
Penelope Semrau

Preparing Computer Education Leadership
for the 1990's and Beyond
Robert Hilgenfeld

Projects
Teacher Education Science Methods Planning Instruction *76*

An Interactive Video Lesson Planning Project for Teacher Education
Charlotte Scherer

One Banana, Two Banana—
Science and Math with Integrated Applications
Elizabeth Viau

Multimedia in Teacher Education
Ronald J. Abate

Projects
Exciting Projects for Native American Students *78*

A Multimedia Approach to Teaching the Ojibwe Language
Richard Sgarlotti

Recreating Native American Legends with HyperCard
George Beekman

The Native American Educational Network: Its Impact on Teachers and Administrators in Geogrpaphically Isolated Schools
David Basham

Paper
HyperStories: Cultivating Interactive Reading and Writing on the Macintosh Computer *80*
Joyce Brody and Mary Lange

Paper
Managing Multimedia Applications Using DESQview *85*
David L. Doss and Bill E. Swafford

Special Presentation
Computers for Young Children *89*
Harriet Paige

Special Presentation
The V. I. C. Awards: Video in the Classroom Award Winning Applications *89*
Hall Davidson

Projects
Research—Computer Findings *90*

Educational Computing:
A Success Story or an Expensive Preoccupation
Andrew Tellep

University of Northern Iowa Computerized Middle School Curriculum Center
Philip Nelson

The Evaluation of Interactive Video Instruction
Sandra Levin

Special Presentation
Logo as Intellectual Laboratory and Vehicle for Self Expression *91*
Gary Stager

Paper
Programming Variables: More Than Just a Name *92*
Tronie Gunn

Paper
Teaching the Idealizations: Altering Science Alternative Conceptions with Microcomputer Simulations *97*
Herman G. Weller and Norman R. Dodl

Special Presentation—ISTE
Responsible Computing: Computers, Ethics and Students *103*

Special Presentation—ACM SIGCUE/ISTE
Technology Education for Teachers:
Parts I & II—Focus, Activities, and Direction *104*

Special Presentation—ISTE/SIGTC
The Role of Technology in the Restructuring of Schools *105*

Projects
Hypermedia/Multimedia Applications *106*

Animal Safari: Multimedia Learning for Special Needs Students
Christine Appert

Hypermedia Instructional Enviroments:
Learner Strategies and Learning Outcomes
Kim MacGregor

White Plains Time Traveler—
Local History on Interactive Videodisc
Kathy Rehfield-Pelles and Naomi Gross

Projects
Technology in the Elementary Setting *108*

A New Direction: Distance Education and Chapter 1
Dianna Lawyer-Brook

Interpreting Literature Through Art and Technology:
A Collaborative Project
Barbara Linsley and Joan Hanor

The WHAT'S IN THE NEWS Telecomputing Project
Chris Clark

Projects
Writing Across the Grade Levels *110*

Implementing the Houston Independent School District Writing Labs
Dr. Richard Alan Smith

Developing Creativity in Students through Multimedia
Stephen F. Bombara

An Interactive Writing Program for Urban Teenagers
Randall Ryder

Special Presentation
Information Literacy: A Changing Education Environment *112*

Special Presentation—ACM/SIGCSE
Computer Science: A College Preparatory Course for All High School Students *113*

Projects
Using Computers to Collect, Analyze, and Synthesize *114*

Hands on Data: Putting Kids in Charge of Data!
Phyllis Kalowski

The Tabletop: A Visual Tool for Creating, Exploring, and Analyzing Data
Chris Hancock

Exploratory Data Analysis, Median Polish, and Mathematics
Martin Adickes and Rob Rucker

Projects
"Take a Risk" Students with Special Needs *116*

Enhancing Learning through the Use of Interactive Video
Doris Tyler

Mac-in-Touch
Diane Frank, Carol Goltz, and Mary Margaret Stephens

From Desktop Publishing to Hypermedia
Kathleen Sutphen

Projects
Teacher Education Telecommunications *118*

Using PROLOG and Hypertext in Undergraduate Teacher Preparation
Gary Tubb

Education Students On-Line at Cal State Los Angeles
James H. Wiebe

Table of Contents—Wednesday

Special Presentation
**Education on the Electronic Frontier:
Learning and Teaching on Tomorrow's Networks** *121*

Projects
Project Outcomes *122*

Computers and Classroom Instructional Research
Shirley Becnel

**Technology Now and Tomorrow—
An Interactive Approach to Elementary Science**
Beth Sandez

Indiana's Project 3R's
Julie Poage and Janelle Zorich

Special Presentation
Visual Arts—A New Frontier in Computer Technology *124*
Nancy Scali

Projects
Computers in the Future *125*

Imperative: Technology Training for Administrators
Hilary Cowan

The Florida Science Videodisc Project
Robert Fronk

The School of the Future: Visions for Learning
Gwen Solomon

Projects
Mathematics Applications: K-University *127*

Ease Students into programming with Turbo Hal
John T. Taylor

**Using Hypertalk Scripting to Introduce
Beginning Algebra Concepts**
Barbara Dubitsky and Rita Kabasakalian

**MathTech 128: A Collaborative Project of
PLAYING TO WIN and COMMUNITY SCHOOL
DISTRICT #4 (NYC)**
Andrea Kimmich-Keyser (Playing to Win) and
Gayle Rubenstein (District #4, New York City)

Paper
**K-12 Computer Teachers Literacy Status and Training Assessment in
Illinois** *129*
Luann Stemler and Joaquin Vila

Paper
**Experiences Retraining Elementary and Middle School
Teachers To Teach Computer Programming** *132*
Robert M. Aiken, Nina P. Bhote,
Glenn E. Snelbecker, and Judith D. Wilson

Special Presentation
Technology! Calculus! and Individual Differences! *138*
Brent Wilson and Michel Pitre

Special Presentation—ISTE
**Accreditation Standards for Instructional Computing and
Technology Teacher Preparation Programs** *139*

Special Presentation
**The Texas Center for Educational Technology:
Current and Future Solutions** *140*
Kenneth E. Brumbaugh and James L. Poirot

Projects
Technology and Special Populations *141*

**The CLC Technology Project: Meeting Literacy Needs of
Special Populations**
Jane D. Steelman

Center for Special Education Technology
Susan Elting

I Care; You Care: Together, You Can Do It!
Diana M. Willis

Paper
The Nihongo Tutorial System for Learning Technical Japanese *143*
Nelson K. Leung and Anthony A. Maciejewski

Paper
"What did he say?": Speech Output in Preschool Software *149*
Erik Strommen

Special Presentation
Software—From Research to Development to Publication *152*

Special Presentation—ISTE
Hypermedia in the Classroom *152*

Projects
**History/Social Studies and Hypermedia/Multimedia
or Interactive Media** *153*

Using Hypermedia to Implement Curricular Change
Linda Smith and Sheri Henderson

Civil War Interactive Multimedia Project
Charles White

**Teaching History/Social Science Through Interactive
Technologies**
Nancy Hunt

Special Presentation
Exploring Simulations in Science *155*

Special Presentation—ACM/SIGCUE
Cooperative Learning—Computers and Technology *155*
Gregg Brownell and Nancy Brownell

Projects
Teachers and Telecommunications *156*

CAWP Online: Supporting Professional Growth with Electronic Linkages
Jane A. Rosen

Teacher Inservice Technology Education: A Partnership Approach in California
Craig Blurton

Special Presentation
Virtual Realities: From the Concrete to the Barely Imaginable *157*
Stephen Marcus

Projects
HyperTraining *158*

HyperMedia: A Performance—Based Model of Inservice Education
Marty Karlin

HyperTEACH—A HyperCard Teacher Utility
Michael Land

HyperCard for Elementary and Secondary Education: A Sampler
Kevin Walsh, Connie Gonser, and John H. Strange

Special Presentation
Two International Telecommunication Models: I*EARN (The International Education and Resource Network) and the New York State/Moscow School Telecommunications Project *159*

Paper
Ending the "Blind Leading Blind" Syndrome: Advocating Technologically Competent Administrators *160*
John F. Beaver

Paper
Selecting Personal Computers for a University MicroComputer Cluster: A Multiple Criteria Decision Analysis Approach *164*
Philipp A. Djang

Special Presentation—ISTE
Network 2001: The Future of Telecomputing in Schools *170*

Projects
Multimedia and Science *171*

The Weather Project: A Coast-to-Coast Telecommunications Project
Eadie Adamson and Carol Goodman

Girls, Computers, and Science: Involving Them with Real World Problems
Diane McGrath

ENLIST Micros III: Models for Establishing ENLIST Micros Teacher Centers
James D. Ellis and Phillip G. Goulding

Projects
Lego Logo Projects *173*

A Jump Start for Future Scientists and Future Science Teachers
Kathleen Sullivan

Creating a Cooperative Learning Environment Using LEGO TC Logo
Jo Ann Karaffa

Bridging the Gender Gap: LEGO—Logo Projects including SPACE STATION 2000 Manorama Talaiver

Special Presentation
The Impact of Computers on the Art Curriculum: Three Case Examples *174*

Projects
"Write On" Intermediate Writing *175*

The Mystery Disc: Encouraging Writing in Junior High School
Connie Gonser, Rosemary Lyons, and Sylvia McGann

Teaching and Learning with Technology
Janet Van Dam

Using Hypermedia to Elicit the Stealth Essay
Gary Brown

Special Presentation
Report from the ACM Two-Year College Curriculum Task Force .. *177*

Paper
Using the Scientific Method in Introductory Course Lab Exercises *178*
R. Dale Johnson

Paper
Teaching Twos Complement Arithmetic in the Classroom *183*
Billy Wood

Special Presentation—HyperSIG
Hypermedia in Colleges and Universities: What are We Teaching and How Are We Teaching It? *187*

Paper
Achievement in a Computer-Rich Environment *188*
W. George Cathcart

Paper
ITEC: An International Research Project on Children and Computers *195*
Betty Collis

Projects
Networks *201*

Implementation and Evaluation of an Electronic Network for Teachers and Administrators
Ann Thompson and Julie Hamilton

Using Computer Conferencing to Bridge the Gap Between K-12 and College
William W. Wright, Jr.

The IBM/NEA School Renewal Network: Linking Practitioners and Researchers
Shari Castle

Projects
Gender Equity in Computing *203*

Using Hypermedia to Bring Young Women to Science
Rachelle S. Heller, C. Dianne Martin, and Emad Mahmoud

The Portrayal of Females in Computer Education Texts Grades K-12
Gregg Brownell

Calculating Women: Filling in the Time Line
Marcelle Bessman

Projects
USA Students Exchange with Israel, Haiti, and USSR *204*

Tall Tales: Intercultural Creative Writing Exchange Between Russian and American Students
Anna Gritsouk and Michael Waugh

Putting Life into Literature Via the International Teleclass
Judith Eber

So What Can You Do With Technology and Second Languages?
Ivy J. Montoya

Special Presentation
Long Distance Learning and Computers *206*
Tom Campbell

Special Presentation
Federal Perspectives on Technology and Restructuring *206*

Special Presentation
Interactive Multimedia in the 90s: Developments, Trends, Prospects, and Applications *207*

Special Presentation—ISTE
Out of the Trojan Horse—Logo's Role in Educational Reform *208*

Projects
Computers and Girls *209*

Where Is Carmen? In the Classroom
Mary Jane Ford and Nancy Markwell

Equity in Technology: Removing the Barriers to Equitable Computer Access
Alice Fredman

Girls and Computer Technology— Maintaining an Interest
Joyce Kasten

Special Presentation—ISTE
Technology Makes a Difference: Improving Our Schools *210*
Lud Braun

Projects
International Computing: Mexico *211*

Teaching High School Physics Through the Use of Computer Simulation Laboratories
Enrique Calderon

On New Tools to Close the Gap Between Between Mathematics
Gustavo R. Deffis

Logic and Creativity Experiences at Kindergarten Level Using the "Magic Microwave Oven"
Claudine Court

Discovering Language Structures
Rocio Gonzalez

Special Presentation—ASCUE
How to Approach Knowledge-Based Systems: PROLOG or Expert System Shells? *213*
Waldo Roth and Antonio Lopez

Special Presentation—SCS
Mathematics Exploration and Experimentation: A Technology Based Teacher Development Course for Inquiry Teaching and Learning *213*

Projects
Hypertext/Hypermedia in Content Areas: Music and Business *214*

The FoxWay Project: Hypertext v. Database
Deborah Healeyand and Mark Horney

Integrating MIDI, Videodisc and HyperCard as a Music Educational System
Clinton E. Stapleton and James A. Levin

Hypertext and Professional Communications: Developing Design Principles
Emily A. Thrush

Special Presentation
Media Issues in Telecommunication *216*
Marina Stock McIsaac

Paper
Evaluating Students' Use of Electronic Communication to Build Consensus: A Pilot Study *217*
Ann L. Lyness and Jean A. Raimond

Projects
Video Communications *224*

Student Communications through Multimedia Technology
John Little and Sandee Holiday

Restructured Teaching For Distance Education
Don Knezek

Making Tremendous Videos
Patty Achey-Cutts

Projects
Doing the "Write" Thing *226*

Talking Computers and At-Risk Students
Dorothy Noyce and Cathy Conger

Do the Write Thing—
An Innovative ComputerWriting Project
John Scheiss

Promoting Writing With Primary Students in the
Computer Lab
Patricia Friesen

Paper
Computer Mediated Conferences for Collaborative Learning:
Towards Hypermedium? *228*
A.Derycke, D.Clément, C.Ladesou, and C.Viéville

Paper
Composing Process Software and Writing Instruction *235*
W. Michael Reed

Projects
Music and the Arts *241*

Logo, Art, and Architecture in the Middle School
Chuck Friesen

Enchanted Amiga Room: Programmed Take-Off on
Disney's Enchanted Tiki Room
Maria Weiss

Combining Graphic Arts and Animation
Jim Madeheim

Projects
Multimedia Teacher Education *242*

Extending Multimedia Computing Through Distance Learning
Nancy Dill

Linking Member Groups with ASCD's Electronic Network
Vicki Hancock

A National System for Electronic Transfer of Student Records
Robert Friedman

Projects
Science Across the Levels *244*

Using Computers for Field Work in the Southwest/West
Desert Environment
Jim Anton

Interactive Multimedia Social Science Simulation Project
"Flatnoid"
G. Beytin, M. Elowitz, and L. Elowitz

Projects
Bilingual Computing *246*

Creative Computing with ESL/LEP Students
Sue Espinoza and Judi Repman

Building Connections: Students and Hypermedia
Ann Dana and Marianne Handler

If It's Saturday—It Must Be Computer Check-Out Day!
Diana M. Willis

Projects
Teaching Global Education via Telecommunications *247*

Learning Circle Projects Sponsored by Global Partners
Margaret Riel and Adel Labib

Telecommunications in a Global Community
Virginia Lawson

We Are One World:
A Multicultural Telecommunications Project
Ann Grady

Special Presentation
WAR IN AI: What Does it Mean for Educational Computing? *249*
Seymour Papert

Special Presentation—ECMI
Funding Computing Research and Instruction
at Minority Institutions *250*
Jesse C. Lewis

Special Presentation—ACH
What are Networked Information Resources
and Where to Find Them *251*
George Brett

Table of Contents—Thursday

Special Presentation
The Editing of Reality: Desktop Video & Multimedia Composition as Basic Skills for the 90s *255*
Ted Kahn

Special Presentation
Logo Based Hypermedia Systems for Small Computers *256*
Michael Tempel

Projects
Computer Uses: ESL, CAI and Family Literacy *257*

Using Computers to Teach American Idioms to ESL Students
Mei-Yan Lu

The Use of Word-Processing Macros to Create CAI Routines
Joel Nydahl

Family Literacy and Computers
Rebecca King

Projects
Problem Solving *258*

Kids Today: Adults Tomorrow—
Computers in the 21st Century
Mary S. Haverfield and Brevard Williams

Computer Database Classroom Activites:
Updates From a National Study
Vivian Johnson

The Use of TETRIS in the Mathematics Curriculum
Ray Braswell and Leah McCoy

Special Presentation—SCS
Inquiry Learning in Mathematics *259*

Projects
Teacher Training *260*

Administrative Staff Development
Jeffrey Branzburg

Teachers and Technology: Making the Connection
Carole Martinez and Elizabeth Bankhead

An Integrated Curriculum Model of Computer Education
for Elementary Teachers
Kathleen Skinner and Christine Nelson

Projects
Current K-12 Educational Technology Research Activities *261*

Projects
Computers are Good Listeners: Deaf Education Enrichment *262*

Computer Camp for Communications Exceptionalities
John M. Smith

Computer Networks as Tools for Supporting
Writing Development in Deaf Students
Babette Moeller, Carol Reich, and Laura Bell

"Talk to Me," Kids, Computers, Video for
Whole Language Telecomputing
Debra Freedman

Paper
Using Computers and Telecommunications in the Elementary School *263*
Gloria Cathcart

Paper
Learning Circle Projects Sponsored by Global Partners *270*
Margaret Riel and Adel Labib

Special Presentation—ACM/SIGCUE
Expanding the Base of Education Programs: Computer Based Training in the Corporate Environment *276*
Regan Carey and Doris Carey

Special Presentation
Global Mind Change— The Challenge of Restructuring Education *276*
David Thornburg

Special Presentation
Technology-Based Manipulative Approaches to Teaching Mathematics *277*
Fred Ventura

Special Presentation—ACM/SIGUCCS
NSFNET—A Network for Research and Education *277*
Douglas Gale

Projects
Diagnostic Skills Trouble Shooting *278*

Troubleshooting Complex Aircraft Systems:
A Hypermedia CD-ROM Development Project
Martha H. Weller and Michael R. Michela

Computer Based Error Correction Simulations for
Training English Composition Teachers
Lynne Diaz-Rico

Knowledge-Based Inference Tool for Enhancing
Clinical Diagnostic Skills
Frank J. Papa, Jon I. Young, Gerald A. Knezek and J. Shores

Special Presentation
Logo Action Research Collaborative's National Sites Report 279

Special Presentation
Model Technology Schools Projects: What Have We Learned? 279

Projects
Telecommunications in K-12 Environment 280

Telecommunications Across the Curriculum and Around the Globe
Brevard Williams III

Internet in the High School
Janet Meizel

Creating Interdisciplinary Links Using Computer Teleconferencing
Peter Hutcher

Special Presentation
Arizona Educational Telecommunication Cooperative 281
A. Ashton

Projects
Just Do It! Implementing Technology 282

How we Got to 'Yes' or Surmounting $1.6 Million in Hard Feelings
Richard Rose

Solution: Project DEIKAS
Jerry Christy

A Successful Model: State-of-the-Art Teaching Using an Integrated Learning System
Debora Doucette

Projects
World and Local Communities 283

Parent Night for National Geographic Kid's Network
Jill Greiner

A Progress Report From the Marin School for Integrated Studies
Karen Randlev

The Initial Charting of Computer Assisted Curriculum
Robert Hess

Projects
How to Make Multimedia Come Alive 284

Multimedia in Maroa? Myth or Magic?
Kathy Bjelland

Improving Instruction in Middle School Mathematics: Multimedia Teacher Training Project
Joyce Friske

Making Your Curriculum Come Alive With Multimedia: Experiencing Multimedia's Magic!
Lynn Lieberman

Projects
Logo Technology 285

Team Logo Meets Currie and Leitner
Camille Minichino, Stephen Sesko, and Brian Webb

Turtles and Toys: Technology in the Classroom
Deborah Bauder, Mary Planow, and Ronald Sarner

What Personality Information Do Computer Facilitated Children's Drawings Reveal?
Judi Harris

Projects
Apple II Applications 287

Bucks Facts, Inc.
Doug Lehnen

Apple IIs: The Most Ignored/Forgotten Hypermedia Tool
Richard Daughenbaugh

Technology for the Asking
John Hernandez

Projects
Higher Level Thinking Skills and Problem Solving 288

Computers for Higher Level Thinking and Cooperative Learning
Marlene Riley and M. Rogers

Process Highlighters in a Computer Simulation: Facilitation of Theory-Oriented Problem Solving
Terrell N. Chandler and Christine Chaille´

Investigating Problem Solving Strategies of Middle School Students Using Computer Software
Barbara Biglan

Projects
Multimedia Faculty Development 289

Multi-Media Instructional Products
John H. Strange, Lora Kawka Davis, and Jennifer Smith

Transforming Instructional Strategies into Computer Courseware
Annette Lamb

The Multi-Media Cart: Getting Teachers To Use Technology in Teaching
John H. Strange

Paper
CompuPASS: Computer-Based Assessment and Remediation of Academic and Test-Taking Skills 290
Vera T. Gershner and C. Sue McCullough

Paper
Teaching Cognitive Monitoring Strategies Through Guided Logo Instruction 295
Mi Ok C. Lee and Ann D. Thompson

Projects
Looking into the Future *315*

Increasing Technological Skills through the Training of Teacher Leaders
Madeleine Long

Taking the Next Steps...
Martha Hancock

Virginia's Plan: A Blueprint for the Present and the Future
Helen Edens

Projects
Special Learners *316*

Integrating Computers into a Communication Disorders Curriculum
Paula Cochran

Use of the Macintosh as an Augmentative Communication Device
Ray Braswell

Computer Access for Visually Impaired College Students
Janet Smith

Projects
Global Classroom Using Telecommunications *317*

Teleconnected Cultures: An International Telecomputing Classroom Project
Bonnie Price

A Telecommunication Tale of War-Time Wonders
Gideon Goldstein and Jeffrey Golub

Special Presentation
Learning Research Institute *318*
Ted Kraver

Special Presentation
Exemplary and Typical Computer-Using Teachers *318*

Special Presentation
Beyond Multimedia to the Virtual Classroom *319*
Fred D'Ignazio

Paper
Technology and Teacher Autonomy: Findings of an In-School Study *320*
Peter Gray-Whiteley

Paper
Solutions: Just A Matter of Time... An Effective Staff Development Model for Follow-up and Continuing Assistance in Technology Integration *325*
Kam Matray

Projects
Multicultural Computer Projects *329*

Computerized Spinoff: A New Look at South Philadelphia High School in the 1990s
C. Tina Petrone, A. Indictor, E. Kelberg, and S. Williams

Animation Is a Second Language
Joyce Morris and Carol Adamsky

HyperText Folklife Encyclopedia and a Community-Based K-9 Whole-Life Curriculum
Kathleen Duplantier

Projects
Metacognition *330*

Helping High School Freshman Develop Transferable Computer Skills
Eric K. Neufer

Model Technologies School Project
Ronald Rescigno

Teaching Thinking Skills To All Children
Carol E. Edwards

Special Presentation
The Required Computer Course for Education Majors: 1991 Update *332*

Projects
Curriculum Integration *333*

Transforming Instructional Strategies into Computer Courseware
Annette Lamb

A Computational Tutor for Architectural Design
Patricia Macintosh

Two to Tango-The School-College Dyad in Promoting Technology
M.S. Vijay Kumar and William Clark

Special Presentation—ISTE
Telecomputing Networks: A Tool to Link Educators *334*

Projects
Telecommunications and Online Databases Across the Curriculum *335*

Information Skills in the K-8 Curriculum
Sandy Lobeck

Using Databases to Teach Research Skills to ESL Students in Freshmen Composition
Beverley Crane

The Electronic Research Center Project
Ann Lathrop

Paper
A Case Study for Developing Hypermedia Software *336*
Stacie Hibino and Kathleen Brade

Paper
A Case Study for Developing Hypermedia Software *336*
Stacie Hibino and Kathleen Brade

Paper
Teaching Business Policy Using Multimedia Game Simulation Support *341*
Richard Cobb

Projects
Integrating Laser Disks in Your Classroom *345*

Interdisciplinary Instruction Using Laser Videodisc
Jim O'Brien

Portable Multimedia Unit Project Report
Gerald Williams

Repurposing Video and Audio Discs for Interactive Classroom Use
James Wall

NECC '91 Conference Committees

NECC '91 Conference Committee

Gary G. Bitter
Conference Co-Chair
Arizona State University

Mary M. Hatfield
Conference Co-Chair
Arizona State University

Helen L. Padgett
Program Chair, Papers
Arizona State University

Janie Wilson
Local Arrangements Chair
Arizona State University

Anita Best
ISTE Representative to Program Committee
ISTE

Susan Gayle
Conference Coordinator, Proceedings
ISTE

Clifford Hurston
Gloria N. Wilson
Equipment
Arizona State University

Martha Rader
Treasurer
Arizona State University

NECC '91/Arizona State University Registrar
Arizona State University

Warren Gamas
Social Events
Arizona State University

Wayne Hartschuh
Social Events
Arizona State University

Barb Coady
Projects, K-12
Peoria School District

Jane Davis
Projects, K-12
Peoria School District

Norman Higgins
Projects, University
Arizona State University

Maurene Gerson
Workshops
Arizona State University

Janine Muto
Society Sessions, Birds-of-a-Feather
Chandler School District

Roger Yohe
Special Sessions
Estrella Mountain Community College

Mike Charles
Presiders
Pendergast School District

Local Arrangements and Program Activities

Michelle Bailey
Arizona State University

Anna-Lisa Byrne
Glendale Elementary District

Allan Cameron
Phoenix Union High School District

Barbara Clark
Arizona State University

Chris Cordova
Arizona Department of Education

Joe Demko
Phoenix Union High School District

Kitti Leslie
Glendale Elementary District

Jennifer Mayhew
Arizona State University

Barbara Menzel
Arizona State University

Elaine Phares
Arizona State University

Rocio Sanchez
Arizona State University

Paul Skiera
Arizona State University

1990–1991 NECC Steering Committee

Ronald E. Anderson
ACM/SIGCAS
University of Minnesota

Gary G. Bitter
Conference Co-Chair, NECC '91
Arizona State University

David Brittain
ISTE
Florida Dept. of Education

George Brett
ACH
University of North Carolina

Della Bonnette
Vice Chairperson
University of SW Louisiana
Steve Gilbert

EDUCOM
Washington, DC

Diana Harris
Secretary
University of Iowa

Mary M. Hatfield
Conference Co-Chair, NECC '91
Arizona State University

James Kerlin
ACM/SIGUCCS
Pennsylvania State University

John Lawson
ACM/SIGCUE
Lewis Clark State College

Jesse C. Lewis
ECMI
Norfolk State University

Doris K. Lidtke
Past Chairperson
Towson State University

Bonnie Marks
ISTE/SIGTC
Alameda County Office of Education

John D. McGregor
Conference Chair, NECC '90
Clemson University

NECC '91 Conference Committees

David Moursund
ISTE Representative to
Exec. Committee
ISTE

Mike Mulder
IEEE Computer Society
University of SW Louisiana

Cathleen A. Norris
Conference Chair, NECC '92
Univ. of North Texas

Helen L. Padgett
Program Chair, NECC '91
Arizona State University

Jim Poirot
Chairman
University of North Texas

Nancy Roberts
SCS
Lesley College

Wally Roth
ASCUE
Taylor University

Ted Sjoerdsma
Treasurer
Washington and Lee
University

Helen C. Takacs
Program Chair, NECC '90
Mississippi State University

Harriet Taylor
Program Chair, NECC '92
Louisiana State University

Joe Turner
ACM/SIGCSE
Clemson University

Cooperating Societies

ACH—
Association for Computers
and the Humanities

ACM—
Association for Computing
Machinery

SIGCAS—
Special Interest Groups on:
Computers and Society

SIGCSE—
Computer Science Education

SIGCUE—
Computer Uses in Education

SIGUCCS—
University and College
Computer Services

ASCUE—
Association for Small
Computer Users in Education

ECMI—
Educational Computing at
Minority Institutions

EDUCOM

IEEE-CS—
IEEE Computer Society

ISTE—
International Society for
Technology in Education

SIGTC—
Special Interest Group for:
Technology Coordinators

SCS—
Society for Computer
Simulation

Society Descriptions

ACH—Association for Computers and the Humanities

The Association for Computers and the Humanities (ACH) is an international organization devoted to encouraging the development and use of computing techniques in humanities research and education. Traditionally, ACH has fostered computer-aided research in literature and language, history, philosophy, anthropology, and related social sciences as well as computer use in the creation and study of art, music, and dance. As computing applications in the humanities have developed and broadened in the 1980s, the Association has expanded it scope to include areas from word processing to computer-assisted instruction in composition, language, history, philosophy, and anthropology, as well as computational linguistics and cognitive science, which overlap increasingly with work in the area of humanities computing.

Founded in 1977, ACH is the primary professional society for humanists who are involved or interested in any aspect of humanities computing. The Association provides a forum for continuing communication about humanities computing and strives to meet the needs of those who want to gain familiarity with both existing and potential applications of computers in humanities disciplines.

The heart of ACH is its quarterly newsletter, which covers the activities of the Association and its members and includes articles on various areas within humanities computing, news of projects and conferences of interest to ACH members, and reports on activities of governmental agencies and other organizations that affect computer-aided humanities research.

Computers and the Humanities, published by Paradigm Press, is a quarterly journal devoted to scholarship in the field of humanities computing. Subscription to *CHum* is included in the ACH membership fee.

ACH sponsors the biannual International Conference on Computers and the Humanities (ICCH), held in odd-numbered years, which brings together scholars from around the word to report on research activities and software and hardware developments in the field. Recently, ACH began to sponsor conferences and workshops on specialized topics in humanities computing, held in even-numbered years.

SIGCAS—ACM Special Interest Group on Computers and Society

SIGCAS is the Association for Computing Machinery's Special Interest Group on Computers and Society. With a membership of nearly 1,200, this professional group seeks to identify social issues raised by computer technology and to provide a forum for discussion on how to approach these issues.

SIGCAS publishes a quarterly newsletter, *Computers and Society*, which is a primary source of material on this topic. As a vehicle of communication for the SIGCAS membership, it includes news, comments, and articles on any societal issues raised by computing technology. One of the few periodicals on this subject, it provides a flexible and timely forum for important, evolving topics, such as data quality, employment, and intellectual property rights.

According to a recent membership survey, about 40% of SIGCAS members teach course material on computers and society. In recent years SIGCAS has organized sessions at computer conferences on topics such as computer ethics and organizational impacts of computers.

SIGCSE—ACM Special Interest Group on Computer Science Education

SIGCSE became a special interest group of ACM in 1970. It currently consists of over 2000 members from the educational, industrial, and governmental communities interested in various aspects of computer science education. SIGCSE has goals of encouraging and assisting in the development of effective academic programs and courses in computer science and promoting research in computer science education.

The following are objectives of SIGCSE:

1. To provide a continuing forum for discussion of common problems among education and other computer scientists through organized meetings and symposia.
2. To publish a bulletin at least quarterly containing information aimed specifically at those interested in computer science education.
3. To work closely with the Education Board of ACM to insure implementation of effective education programs by the Association.

SIGCUE—ACM Special Interest Group on Computer Uses in Education

SIGCUE provides a forum for the discussion of ideas, methods, and policies related to all aspects of computers in the educational process. Established in 1969 its membership (over 1,400 persons) comes from many countries and numerous, diverse institutions and businesses.

SIGCUE publishes a newsletter titled the SIGCUE *Outlook*. Recent topics have included Preservice Education in Educational Computing, International Reports on Educational Computing, and a Teacher Training Curriculum Project. SIGCUE also sponsors and organizes technical sessions at ACM annual meetings, the National Educational Computing Conference, and other national and regional meetings of interest to its members.

Among SIGCUE's goals are (1) helping to bring the technical expertise within ACM to bear upon educational compting generally, (2) cooperating with other special interest groups or educational societies to promote attention to educational computing issues, and (3) providing written and verbal forums for members and the educational community to exchange ideas concerning computer uses in education.

SIGUCCS—ACM Special Interest Group on University and College Computing Services

SIGUCCS provides a forum for those involved in providing computing services on a college or university campus. The topics addressed by SIGUCCS include managing campus computing, computing as it relates to the overall goals of the institution, and the state-of-the-art in various types of college and university computing services, and provides opportunities to discuss and share ideas and experiences with others.

Two annual conferences are regular activities of SIGUCCS. The Computing Center Management Symposium addresses the many aspects of managing computing on campus. This includes hardware, software, planning, finances, and personnel, to name few. The User Services Conference deals more directly with the delivery of particular services to the higher education community. Tutorials on relevant issues are held at both conferences.

In other projects, SIGUCCS offers a Peer Review of the university computing function. Upon request of the computer center director, members of SIGUCCS will formally analyze and comment on different areas of the campus computing function. SIGUCCS also publishes a quarterly newsletter. We consider the newsletter our most important form of communication as it reaches all members and is subscribed to by numerous university computing centers. Conference proceedings are published either as separate documents or as part of the newsletter itself.

ASCUE—The Association of Small Computer Users in Education

The Association of Small Computer Users in Education (ASCUE) is a group of small colleges who work toward continued quality computer education and computer assistance in both administrative and academic areas. The purposes of the organization are: 1) to encourage appropriate uses of computing equipment and techniques for the improvement of its member institutions; 2) to supply its members with information on the most current computing methods; 3) to assist them in solving individual problems; 4) to cooperate with them in the utilization of the various small computers; 5) to cooperate with manufacturers, distributors, and suppliers in establishing and maintaining proper technical standards and in meeting new needs for special devices and systems.

ECMI—Educational Computing in Minority Institutions

ECMI is an organization run by a steering committee representing institutions whose student body reflects a large identifiable minority population. The objectives of ECMI are:

1. Computer literary: To create among the faculty and administrators of the minority institutions an awareness and understanding of the strengths and weaknesses, uses and issues, advantages and disadvantages, feasibility, practicability and limitations of computer applications in all aspects of society, including education.
2. Educational Computing: To narrow the gap which exists between the faculties in minority and non-minority institutions with respect to educational computing know-how and access.
3. Research Computing: To improve the computing facilities available to faculty of minority institutions for research purposes, particularly in those institutions offering graduate programs.
4. Technical assistance—consultants: To provide expert and impartial technical assistance to academic administrators of minority institutions on all phases of academic computing (Instruction and research).
5. Education programs in the computer sciences: To improve the offerings of courses and degree programs in the computer sciences at minority institutions at all levels (e.g., introductory courses, minors, 2-year degree programs, 4-year degree programs, continuing education, graduate programs).
6. Computing facilities: To improve both quality and quantity of computing facilities available in minority institutions, because experience in the non-minority institutions has shown that an adequately staffed and equipped computer center for academic computing is essential to the success of previously stated objectives.
7. Direct student assistance: To increase the availability of minority staff for the computer centers and computer science education programs of minority institutions.
8. The need for a comprehensive program: To facilitate co-ordination and equitable distribution of funded activities to qualified institutions, associations, etc.

EDUCOM

EDUCOM is a nonprofit consortium of colleges, universities, and other institutions founded in 1964 to facilitate the introduction, use, and management of information technology. Through direct services and cooperative efforts, EDUCOM assists its members and provides leadership to the higher education community.

EDUCOM is funded by membership dues, service fees, and grants from foundations, corporations, and government. EDUCOM activities involve over 530 institutions and 90 corporations in the U.S. and abroad. In brief, they are: EDUCOM General Membership; the EDUCOM Consulting Group; EDUCOM Networking Activities including the Networking and Telecommunications Task Force (NTTF) and BITNIC, the BITNET Network Information Center; the EDUCOM Software Initiative; and the Corporate Associates Program.

IEEE—The IEEE Computer Society

The Computer Society is the world's largest association of computing professionals, with a total membership of approximately 110,000 computer scientists, computer engineers, and interested professionals. Society membership is open to IEEE members, associate members, and student members and to non-IEEE members who qualify for affiliate membership. An affiliate member is a person who has achieved status in his or her chosen field of specialization and whose interests focus in the computing field.

Every Computer Society member receives *COMPUTER*, a peer-reviewed monthly magazine of general interest to computing professionals which also covers society news and events. Five specialized magazines and five journals are also available to society members as optional subscriptions and to nonmembers, libraries, and organizations.

Magazines published by the Computer Society include *IEEE COMPUTER GRAPHICS & APPLICATIONS*, *IEEE MICRO*, *IEEE DESIGN & TEST*, *IEEE SOFTWARE*, and *IEEE EXPERT*. Research-oriented journals include *IEEE TRANSACTIONS ON COMPUTERS*, *IEEE TRANSACTIONS ON SOFTWARE ENGINEERING*, *IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE*, *IEEE TRANSACTIONS ON KNOWLEDGE AND DATABASE ENGINEERING*, and *IEEE TRANSACTIONS ON PARALLEL DISTRIBUTED SYSTEMS*. The Computer Society Press publishes nonperiodical literature, including tutorial texts and conference records. The society's catalog contains approximately 900 titles which are available for purchase.

The society sponsors or cosponsors more than 100 conferences and meetings ranging from workshops and symposia with a few dozen participants to major conferences with many thousands of attendees. Over 30 technical committees offer the opportunity to interact with peers in technical specialty areas, receive newsletters, and conduct conferences and tutorials.

The Computer Society has over 100 local chapters throughout the world, and an additional 100-plus student chapters which provide the opportunity to interact with local colleagues and hear experts discuss technical issues. In addition, tutorials, educational activities, accreditation of computer science and engineering academic programs, the development of standards, and an international electronic mail network all play prominent roles in the society's activities.

ISTE—International Society for Technology in Education

The International Society for Technology in Education, ISTE, is the merged society of the International Council for Computers in Education and the International Association for Computing in Education. ISTE is a non-profit educational organization, with 12,000 individual members and over 60 organization and associate members.

ISTE publishes *The Computing Teacher*, *Journal of Research on Computing in Education*, *CÆLL Digest*, *Microsoft Works in Education*, and books and courseware for persons interested in the instructional use of computers at the precollege level. These publications emphasize teaching about computers, teaching using computers, teacher education, and the impact of computers on curriculum.

ISTE has a substantial and growing professional outreach program. Five major components currently include:

1. Organization Affiliate Members. Members publish newsletters and/or journals, hold conferences, and directly interact with their own members.
2. Professional Staff. ISTE has a professional staff who write, edit, participate in conferences, process orders, consult by phone or mail, etc. *The Computing Teacher* is put together by a full in-house production staff.
3. Ad Hoc Committees. Such a committee created the "ICCE Policy Statement on Software Copyright" and "Code of Ethical Conduct for Computer Using Teachers."
4. Special Interest Groups. ISTE has organized special interest groups for computer coordinators, teachers of educators, computer science educators, Logo-using educators, telecommunications, and hypermedia/multimedia.
5. Independent Study courses. ISTE offers eleven independent study courses carrying graduate credit from the Oregon State System of Higher Education.

SCS—The Society for Computer Simulation

SCS is the only technical society devoted primarily to the advancement of simulation and allied technology. It has a worldwide membership and a network of regional councils that covers the United States, Canada, and the United Kingdom.

Simulation is used in every scientific and technical discipline including engineering, manufacturing, biomedical, business, and aerospace. Artificial intelligence, robotics, and CAD/ACM and simulators are areas that have been specifically recognized as important to SCS members.

There are three major SCS conferences each year. They are the SCS Western Multiconference, the SCS Eastern Multiconference, and the Summer Computer Simulation Conference (SCSC).

SIGTC—The Special Interest Group for Technology Coordinators

The Special Interest Group for Technology Coordinators (SIGTC) is a professional organization that helps technology coordinators meet the challenges of a rapidly changing field. We provide an excellent forum to identify problems and solutions, and share information on issues facing technology coordinators at the precollege level.

SIGTC publishes *SIGTC Connections*, a quarterly publication, through the International Society for Technology in Education (ISTE). Articles in *SIGTC Connections* contains helpful information and answers to questions such as:

- Who's doing what to organize and communicate with teachers and administrators?
- What committees and techniques work best for other coordinators?
- How can technology coordinators enlist the support of school site administrators?
- How do others disseminate information?
- What are some successful models for providing staff development?
- How should buildings and districts establish and evaluate the use of educational technology in their programs?

SIGTC organizes a NECC workshop for technology coordinators each year and holds its annual business meeting at NECC.

Reviewer Acknowledgements

The National Educational Computing Conference thanks the following people for their contribution of knowledge, effort, and time as referees for the papers submitted for presentation.

Eadie Adamson
Robert Allen
Lynne Anderson-Inman
Richard Austing
Bruce Barnes
George Bass, Jr.
F. Samuel Bauer
James Beck
Danielle Bernstein
Anita Best
Roy Bhagaloo
Rick Billstein
William Blubaugh
Della Bonnette
Barbara Boyer
James Bradley
David Brittain
J.S. Cameron
Pat Campbell
Lowell Carmony
Cathy Carney
N. John Castellan, Jr.
Margaret Christensen
Betty Collis
Leslie Conery
Sheila Cory
George Culp
Herbert Dershem
Mike Dunlap
Linda Ettinger
Reynolds Ferrante
Frederick Fink
Dexter Fletcher
Larry Flick
David Flowers
Judith Gerstig
Harold Grossman
Dale Hanchey
Cindy Meyer Hanchey
Seymour Hanfling
Clifton Harris
Janet Hartman
David Hata
Sandee Hedetniemi
Henry Heikkinen
Sam Hicken
Mark Horney
Carolyn Jacobi
Betty Jehn
Lawrence Jehn
Dale Johnson
Norm Johnson
Tony Jongejan
Ted Kahn
Peter Kelman
Henry Kepner, Jr.
Joyce Killian
Carolyn Knox-Quinn
D. Midian Kurland
Barbara Kurshan
Annabelle Lavier
John Mark Leach
Laura Leventhal
Mei-Yan Lu
William Lyle
Harold MacDermot
Gary Marchionini
Stephen Marcus
Kathleen Maury
Donald McClain
Sharon McCoy Bell
Diane McGrath
Bonnie Mohnsen
Andrew Molnar
Adeline Naiman
Maggie Neiss
Larry North
Jason Ohler
Trevor Owen
Janet Parker
Jim Parry
Teri Perl
Linda Petty
Barry Pitsch
Richard Plishka
Linda Polin
Gerald Pollard
Doris Ray
Dick Ricketts
Nancy Roberts
Doug Rogers
Marian Rosen
Donald Ross
Charlotte Scherer
Lynne Schrum
Robert Seidman
Penelope Semrau
Stephen Sesko
Robert Slotnick
Dennis Spuck
D. Stevenson
Neal Strudler
Macey Taylor
Harriet Taylor
John Tenny
Leslie Thyberg
Sam Tumolo
V. Lynne Tyler
John Vaille
Molly Watt
Daniel Watt
John Wedman
David Weinman
Billy Wood
Mary Yates
Shelley York Rose

Special Presentation
Interactive Videodisc Technology in Science Instruction: Matching Methods and Materials

Thomas E. Boudrot
Educational Consultant
Brooklyn, New York

In this one-hour session, participants will be directly involved in an Interactive NOVA demonstration lesson that models exemplary methods and materials in science education. Using a single videodisc player and one Macintosh computer, participants will engage in a cooperative learning experience that explores the concept of food chains and differentiates them from food webs. From the classroom simulation, discussion about effective teaching methods using interactive videodisc technology will ensue.

In this session, participants will:

Envision possibilities for multimedia and videodisc to cross grade levels and meet a variety of student needs and abilities.

Explore strategies for using interactive video disks in a variety of formats ranging from whole class presentation to individual research.

Explore the feasibility of integrating interactive videodisc technology with existing science instruction

Discuss the ramifications upon preservice teacher education and inservice staff development programs.

Projects
Math Applications in K-12 Environments

Can Strategy-Based Computer Instruction Enhance Early Mathematics Understanding?

Douglas Frye
Jodie Ambrosino
Cheng Dong Lu
Computer Science Department
Yale University
New Haven, CT

Abstract

American children relative to their peers in other industrialized countries lag significantly behind in mathematics from the beginning of school. The Kidscount project at Yale has designed a new type of software to improve elementary school children's initial understanding of number, counting, cardinality, addition and subtraction. The software integrates voice, a touchscreen, graphics and animation to produce interactive computer environments that encourage and support the child's discovery of basic cognitive strategies in math. The software is being tested with kindergarten through second grade students in several public elementary schools in New Haven. The results to date show that the strategy-based software produces significantly better math learning than conventional arithmetic software or normal classroom instruction by itself. Because the strategy-based software is clearly more effective than conventional arithmetic software, the improvement in learning is related to the software's emphasis on strategies rather than just being a consequence of computerized instruction alone. Our results also show that the gains in the strategy-based group transferred from the computer to the more usual presentation of arithmetic problems on paper. The findings indicate that a strategy-based approach leads to a new form of educational software and a means of improving early math learning in school.

Multi-sensory Manipulatives and Developmental Mathematics

David Fowler
UNL Teacher's College
Lincoln, Nebraska

Abstract

This demonstration includes sound as a manipulated medium. The concept of "manipulation" is extended to include computer experimentation with "high-tech" programs such as Farallon Computing's SoundEdit program and the symbolic mathematics program Mathematica. It also includes "low-tech" experiments such as throwing a stone into a pond, shaking a rope, and drawing a bow across a violin string. These experiments are interesting and accessible to students over the cognitive developmental range from concrete to formal thinking.

The intent is to have students develop rich sets of mental models across different sensory modalities—visual, auditory and kinesthetic. The repertoire of technical tools available to a teacher allows her to design experiments from which mathematics and correct mental models emerge simultaneously. The use of symbolic notation is then motivated by the need to describe these phenomena more exactly.

Correlated payoffs of this type of instruction include "mathematics across the curriculum," in this case, music, science, and psychology.

Producing Aviation Videodisc Lessons for Kindergarten Math in a Bilingual Environment

Nancy Nelson Knupfer
Arizona State University

Abstract

The kindergarten teacher who is the subject matter expert for this project is actively involved in using an aviation theme in her classroom to motivate the students and to stimulate their learning beyond that of the usual kindergarten curriculum. Last year she planned and conducted an aviation day in which the entire elementary school participated in various events. These events spanned all of the grade levels (K-6) and all of the subject areas in the curriculum. The entire school staff was involved in some way; even the janitors participated in an egg dropping contest in which eggs were dropped off of the school roof to test their structural soundness. Students participated in various demonstrations and contests, there were display booths manned by several prominent men and women pilots and members of the community. Members of the U.S. Air Force, the National Guard, the 99s (a women pilot's organization), the aviation industry, and the community were represented. A news-weather helicopter crew and a crop-dusting helicopter crew landed their helicopters on the school grounds and spoke with students. From the first day of school, the teacher created activities for students to build toward the participation in the aviation day, which occurred during the last month of school. This year the teacher hopes to expand on aviation day and to use technology within the classroom more than before.

Special Presentation—ISTE

Electronic Studying: Computer-Based Outliners Promote Academic Success Across the Curriculum

Lynne Anderson-Inman
Mark Horney
University of Oregon

This presentation describes the results of a federally funded project to explore the use of computer-based outlining programs as study tools. Outlining programs have the potential to enhance reading comprehension, academic achievement, and study skills by (a) increasing student motivation, (b) facilitating meaningful interaction with textbooks, and (c) supporting independent efforts to study content area material. This presentation will describe a three step study process using outliners, share strategies for integrating outliners into the curriculum, and provide data from our implementation efforts in grades 6-12.

Projects

Projects in the Secondary Setting

Computer Magnet School: A High School Model

Pat Hart
Computer Magnet School, AVHS School District
Lancaster, CA

Abstract

Business, CADD, computer applications, electronics, graphic arts, and programming curriculum for the high school level have been merged with computers and brought together into a computer magnet school. The multistrand curriculum is designed to meet the needs of college bound as well as vocational students. The five labs combine Macintosh, IBM, and Apple computers along with laser printers, plotters, scanners, laser disk and CD-ROM players and graphic tablets. The eight member staff of the Antelope Valley Computer Magnet School will share with you software, hardware and curriculum ideas that you can use in your classroom or school.

Changing the Dominate Paradigm—Project 2001

Cathy Cordes
Project 2001 Administrator
Bedford, MA

Abstract

The Bedford Public Schools created the 2001 Committee in January 1990 and charged it with the task of exploring the challenges and opportunities presented by the ongoing development of technology, and defining a shared vision of the Bedford school system of the Year 2001. This vision was to be comprehensive enough to address:

1. the application of technology to the day to day operational aspects of teaching, learning and administration.
2. the impact of technology on curriculum goals and content.
3. the opportunity presented by technology for new learning activities and the potential for enhancing the learning atmosphere.
4. relating themes such as critical and creative thinking, interdisciplinary approaches, cooperative learning, peer coaching, group dynamics skills, global understandings, performance-based assessment of student progress and any necessary restructuring.

This project is distinguished in two major ways. First, we have addressed the type of educational system we want and then fit the technology to it. Second, all 170 professionals of the school system have been involved in the process. The committee did not work in isolation.

This project presentation will summarize the process used by the 2001 Committee internally as well as the process used to involve the entire faculty. Attendees will receive copies of the initial mission statement, timeline for the activities, a summary of the process used, the final vision statement and the outline of the implementation plan. They will also participate in a short activity. The model presented is a valid process for other systems to duplicate.

New Mexico Education Technology Institute

Kurt Steinhaus
Planning Council Coordinator
New Mexico State Department of Education

Abstract

About a year ago, as the National Business Roundtable contact for New Mexico, Mr. Bob Price, retired chairman of the board and chief executive officer of Control Data Corporation, began to work with Governor Carruthers and his Education Cabinet Council to explore new directions for public-private partnerships in New Mexico education. The initial result of these discussions was the first Education Technology Institute held August 6-10, 1990 at five sites throughout the state. Institute participants received five intensive days of training in classroom computer technology, the cost underwritten entirely from private sources. Mr. Price and Control Data became the chief underwriters of the Institute. Once the level of funding for the Institute had been established, solicitations were made from school superintendents for their participant nominees. Preference was given to school districts already involved in the school restructuring process, primarily Re:Learning. Ultimately, one hundred and ten teachers and other school personnel were selected to receive the training.

Paper
A BASIC Programming Curriculum for Enhancing Problem-Solving Ability

Richard C. Overbaugh
West Virginia University

Abstract

Current research shows development of problem-solving skills from learning to program in BASIC. The successful treatments were based on contemporary problem-solving theory, top-down, modular programming and rigorous length and intensity. The proposed BASIC curriculum, incorporating these characteristics, is designed for secondary education and is adaptable to post secondary education. The curriculum suggests beginning with an optional six-week problem-solving software unit to overcome computer anxiety and introduce problem-solving theory. The major portion—BASIC programming—lasts 30 weeks with daily lessons, but is easily expanded or condensed. A brief overview of each unit is given ,and a sample unit is presented in an appendix.

Research and Theory

Microcomputers were introduced to the public as a machine capable of extracting extraordinary intellectual powers from school children. They were supposed to lead to higher thinking skills and increased facility in problem solving abilities. To accomplish this amazing feat, prolific amounts of software appeared for computer aided instruction (CAI) and children of all ages were suddenly programming computers in languages such as Logo and BASIC.

The educational systems were quick to jump on the computer bandwagon because it was purported to be the answer to the inadequacies of American education. Unfortunately, the promises of the computer were not delivered, resulting in a backlash against computers in the early 1980s. In an effort to defend the use of computers in the schools, there was a rush to provide empirical evidence of the positive cognitive effects of programming language instruction, resulting in a large amount of research ranging from no effects to fantastic effects. In general this research was hastily thrown together and contained faulty methods and conclusions. Some of the major inherent problems were: (a) lack of sufficient exposure to the language, (b) lack of control groups, (c) lack of measures tailored to the computer environment, (d) insufficient instruction, or whether or not students really learned a programming language, and (e) computer anxiety (Palumbo & Reed, 1988; Reed & Palumbo, 1988; Reed, Palumbo & Stolar, 1988).

However, it is still currently believed that the teaching of programming languages promotes certain useful skills. Researchers have now had the opportunity to look at and compare earlier work and have begun new research which takes into account the aforementioned problems. This paper looks at the work of professional educators who believe in and are dedicated to the positive aspects of computers in the schools.

What do these professionals think computer programming will do for students? *The first major prospect of learning to program* is the utilization and exercising of higher-level thinking skills to enhance problem-solving ability. Problem-solving is broken down into four phases: specification, planning, coding and debugging (Dalbey & Linn, 1986; Dalbey, Tourniaire & Linn. 1986; Kurland, Pea, Clement & Mawby, 1986; McCoy & Orey, 1988; Reed & Palumbo 1988).

The first phase is *problem specification* which is determining exactly what the problem is and what the program needs to do. Students must evaluate the problem very carefully to know how to start the programming task. Their evaluation should involve questioning, clarifying the outcome of the program, and exploring the different ways to achieve the outcome (Dalbey, Tourniaire & Linn, 1986).

The second phase is *planning*, sometimes called procedural reasoning and is often the most difficult step for students. In this phase, the problem is broken down into the steps that will actually be coded. To do this, the programmer must understand how commands are interpreted by the computer in order to correctly organize the steps of the program. It is widely accepted that programs should be designed using the top-down approach—a process beginning with problem specification, followed by successively breaking the problem down into levels of component parts (Clement, Kurland, Mawby & Pea, 1986; Dalbey & Linn, 1986; Dalbey, Tourniaire & Linn, 1986). At each level is a set of unique algorithms leading progressively to the final algorithms which are then coded, creating the program.

The third phase is *coding*—the actual conversion of the algorithms into a specific language. Not only does a programmer need a large knowledge base of the programming language commands, but also efficient programming skill which is characterized by the use of templates (Dalbey, Tourniaire & Linn, 1986; Magliaro & Burton, 1988). Templates are groups of commands or chunks of programs that students have used to solve previous problems that can be incorporated into the solutions of new problems.

The last phase of programming is *debugging*—the task of finding the errors in a program that prevent its proper execution. Programming errors fall into two general categories: (a) syntax—usually typing errors, and (b) logic and wrong-command errors. Syntax errors are usually easy to locate and correct, but logic errors are much more difficult. To locate a logic error, the programmers decenter, which is a process of separating the function the programmer intended the computer to do from what the computer actually did by analyzing the feedback given by the computer. Becoming proficient in debugging requires a large amount of experience as it is a highly complex task involving abstract rational reasoning (Reed & Palumbo, 1988).

The second major prospect of learning to program is transference of thinking skills and problem solving abilities to other areas. Transference is the reason for supporting a programming curriculum in the schools: If students become skilled in all phases of computer programming, it is believed that they will be able to apply the same thinking processes to noncomputer problems. The teaching of computer languages to develop these skills is ideal because programming is not content directed as are other disciplines. For example, Social Studies and English have particular constituent parts that must be taught, whereas instruction in programming can be tailored to the task of enhancing problem-solving abilities.

Current research is directed toward showing empirical evidence that learning programming increases problem-solving abilities and transfers such skills to other areas. However, because the high level of programming ability needed for the transference of problem solving skills is not attained with present secondary school instruction (Clement, Kurland, Mawby & Pea, 1986; Dalbey, Tourniaire & Linn, 1986; Kurland, Pea, Clement & Mawby, 1986; Palumbo & Reed, 1988), the bulk of the research reviewed for this paper showed positive results in the enhancement of problem solving ability when the measures and learning environments were tailored to the programming environment (Kurland, Pea, Clement & Mawby; McCoy & Orey, 1988; Palumbo & Reed; Reed & Palumbo, 1988). One exception is research by Soloway, Lochhead and Clement (1982) that showed positive transference of problem solving skills to mathematics, which requires thinking process that are similar to those utilized in programming.

The learning environments for several of the studies were carefully constructed and controlled by the researchers in an effort to overcome, or at least reduce, problems that have led to poor results in the past (Dalbey & Linn, 1985; Dalbey, Tourniaire & Linn, 1986; Palumbo & Reed, 1988; Reed & Palumbo, 1988; Reed, Palumbo & Stolar, 1988). Since the results were positive, the instructional designs used can be considered guidelines for educators involved in teaching programming languages. Three aspects are particularly significant: (a) length and intensity of treatment, (b) top-down method of problem-solving and, (c) computer anxiety.

The first aspect is *length and intensity of instruction*. Programming proficiency requires high levels of skill and cannot be achieved in short periods of time. The research examined here largely involved an instruction period covering at least one semester with daily lessons or an equivalent (Palumbo & Reed, 1988). In view of the positive results, this is a clear indication that a curriculum should last at minimum one semester with daily lessons.

The second major aspect is the method of teaching the language. As reported, a *top-down approach*, using modular program construction is necessary for good planning, efficient program design and template formation (Clement, Kurland, Mawby & Pea, 1986; Magliaro & Burton, 1988; Reed, Palumbo & Stolar, 1988). Instructors should teach their classes in a manner that develops planning skills through algorithms and modular programming. The creation and use of templates can be promoted by using programs or parts of programs written by students in subsequent programming problems (Dalbey, Tourniaire & Linn, 1986). It is here that a great deal of support for the use of Logo or Pascal can be found as they both require procedural or modular programming techniques, as opposed to BASIC which does not. However, it has been found that, if BASIC is taught in a modular form using subroutines, there are no differences in the cognitive outcomes (Reed, Palumbo & Stolar, 1988). Therefore, the consideration of cost enters the picture. Logo and Pascal are expensive and must be purchased for each machine in a classroom, whereas BASIC is supplied with almost all machines.

Some researchers feel that certain concepts of programming, such as loops and conditionals, can be taught before a formal language is actually learned. To accomplish this, one group of researchers used Spider World, a simple program that uses commands similar to a programming language but requires minimum coding skills. After the students had a good grasp of the required procedures, they began learning a formal language with positive results (Dalbey & Linn, 1985).

Debugging skills increase as programmers become more skilled in coding and problem solving (Reed & Palumbo, 1988) and by understanding explicitly the function of commands (Dalbey, Tourniaire & Linn, 1986). This process can be started early by engaging students in the task of predicting the outcome of short programs written by an expert. They can also change certain parts of a program (written by an expert) to produce a different outcome. A secondary advantage of this activity is exposure to well-planned programs written by an expert, thus giving students models of efficient programming and templates they can incorporate into their own work.

The final area demanding attention is *computer anxiety*. It has been found that students who feel anxious toward the machine have more problems learning to program well (Reed & Palumbo, 1988). Efforts to overcome this anxiety should take place at the very beginning of a course and can be expedited with the use of easy, non intimidating software to acquaint students with operating techniques and general uses of the computer. Then, when beginning to program, a logical starting point is graphics (Reed & Palumbo) because graphics are perceived as enjoyable by most, are fairly easy to learn, and give an immediate, visual product.

In conclusion, this research supports the notion of the enhancement of higher-level thinking skills and increased problem-solving ability through the medium of computer programming instruction. Therefore, if the reason for teaching a computer language is the acquisition and development of higher thinking skills, the positive results of this research, conducted under conditions designed to promote these skills, should serve as a guide for curriculum development in this area. The fact that there is a lack of supporting evidence in situations not designed with similar objectives is an indication of the present condition of programming instruction. Most courses are taught by instructors whose training lies in other areas, usually resulting in curricula exclusively teaching how to code simple programs. Finally, that the transference of these abilities to other domains, even in research settings, remains to be shown is not surprising. If a high level of proficiency in programming is a requisite for positive results on measures tailored to the computer environment, a need for even further curricular development is indicated before transfer to other domains will commence. Simply put, more research is needed to develop effective methods of teaching computer programming to specifically address the task of promoting problem-solving abilities. Just as important is the absolute necessity for trained professionals who can implement such a curriculum. Without specific curricula and qualified instructors, the potential for positive cognitive results from learning to program will not be realized.

Application

Learning to program in BASIC increases students' problem-solving ability when the instruction is based on the four components of problem-solving—(a) problem specification, (b) planning, (c) coding, and (d) debugging Other significant characteristics of the successful research designs are: (a) the top-down approach to programming, (b) modular program construction techniques, and (c) a treatment sufficiently rigorous to develop an adequate knowledge base of the language. The BASIC curriculum proposed is a non-content-oriented, cumulative, problem-solving approach that encourages higher-level thinking skills. In general, each successive unit requires or, at minimum, can incorporate the skills gained in the previous unit or units.

This paper will present a general description of each unit of the curriculum including very broad objectives and suggested length. A complete sample unit is included as an appendix. A possible class structure, intended to maximize the desired problem-solving approach, and some ideas about specific course content are also discussed.

The curriculum begins with Low Resolution graphics in the immediate mode of execution to give the students instant reaction to the commands entered. This approach also reinforces the later use of line numbers to delay execution.

1. LO-RES GRAPHICS

UNIT OBJECTIVES: The students will-

- initialize a blank disk to use as a work disk
- use simple commands to draw multicolored pictures on the screen in lo-res graphics.
- learn how to edit source code and what to do when error messages occur.
- SAVE, LOAD, edit and RUN programs.
- understand the difference between immediate mode and delayed mode of execution.

Unit two uses Lo-Res graphics to introduce numeric variables, subroutines, simple FOR/NEXT loops for delay, and animation loops. Unit three is High Resolution graphics which utilizes the same program structure and introduces graphics commands similar to those used in Lo-Res. Animation is included to give more practice with loops. Unit three also introduces embedded DOS commands and then segues into the text unit by using very simple text commands such as PRINT and VTAB.

2. ANIMATION IN LO-RES GRAPHICS *2 Weeks*

UNIT OBJECTIVES: The students will-

- use FOR/NEXT Loops and numeric variables to animate simple one color and multi-color objects across the screen from all directions.

3. HIGH-RES GRAPHICS INCLUDING ANIMATION *5 Weeks*

EMBEDDED DOS COMMANDS
UNIT OBJECTIVES: The students will-

- be able to draw static pictures in Hi-Res graphics demonstrate an understanding of the compatibility of Hi-Res colors.
- animate single and multicolored objects in Hi-Res.
- animate objects on a background drawn in Hi-Res.
- use text commands to introduce a graphics program and embedded DOS commands to load and run the programs.
- use embedded DOS commands to alternate between Hi-Res page 1
(HGR) and Hi-Res page 2 (HGR2).

Unit four, is a relatively brief text unit that introduces the remaining screen formatting commands such as FLASH, HTAB, SPEED= and NORMAL. More work with the text commands occurs in the fifth unit, which introduces string variables and gives practice distinguishing string from numeric variables. The concept of conditional statements for program control is also begun in the fifth unit as students program interactive questions and answer programs. Unit six introduces saving data files on disk.

4. TEXT COMMANDS FOR SCREEN FORMATTING *2 Weeks*

INTRODUCTION TO STRING VARIABLES
UNIT OBJECTIVES: The students will-

- learn some text formatting commands.
- use a delay loop
- use simple text commands to introduce a graphics picture and embedded DOS commands to automatically run it.

5. STRING VARIABLES And NUMERIC VARIABLES *3 Weeks*

TEXT FILES, AND MORE EMBEDDED DOS COMMANDS
UNIT OBJECTIVES: The students will-

- use numeric and string variables to create a question and answer program.
- know the symbols for the four basic math functions.
- use conditional statements for responding to user input and error trapping.

6. TEXT FILES *22-25 Weeks*

UNIT OBJECTIVES: The students will-

- save text to a text file on disk
- read text from a text file on disk

Math is the principal focus for unit seven. The four basic math functions were introduced in unit five, therefore, the purpose of this unit is simply to show how programming can be applied to other areas. Some ideas for useful programs are suggested, but the students should be free to use whatever level math they desire for their programs. All commands and skills learned up to this point can be easily integrated into games or interactive math problem-solving programs.

7. MATH UNIT *4 Weeks*

UNIT OBJECTIVES: The students will-

- know 6 simple math symbols and their order of priority and how to change the priority with parenthesis.
- use the LEN command to create a centering routine.
- use embedded DOS commands to alternate two Hi-Res images on the screen.
- use the RND command to create a number guessing game.
- combine the above commands in a single program to title and present any sort of number guessing game.
- use the INT command to round off decimal numbers.
- create programs to solve geometry problems.

Unit eight begins by introducing nested loops, then addresses creating and saving files of user inputted text on a floppy disk to be retrieved and used at another time. The initial task involves saving string variables in a text file and then reading them back. Next, one dimensional arrays are introduced for data storage and then two dimensional arrays for a data base problem.

8. 1 AND 2 DIMENSIONAL ARRAYS *2-3 Weeks*

UNIT OBJECTIVES: The students will-

- understand nested loops.
- understand and use 1 and 2 dimensional arrays.

The final unit is the culmination of the entire curriculum. The students are to design their own program(s) utilizing as many concepts as possible.

9: PERSONAL PROGRAMS *2-3 Weeks*

UNIT OBJECTIVE: The students will-

- use their knowledge to create applications programs that are of interest or use to them and instructor approved.

Class structure should be in a discussion format as much as possible. Present new ideas and concepts as problems and work to solve them together. Create problems through discussion and supply new commands and techniques as solutions. As the course progresses, the structure of the programs and solutions will become familiar to the students. Make sure the four phases of programming/problem-solving are followed carefully: (a) problem specification, (b) planning, (c) coding, and (d) debugging

The first two, problem specification and planning, are the most critical. A great deal of care must be taken to make sure the students develop good methods for deciding what the programming problem is and planning a logical, workable solution.

The course does not include a unit on writing algorithms or flow charts. The reason for this omission is that individuals usually evolve their own form of problem-solving planning. Therefore, forcing them to utilize one particular method may inhibit individual development. In this course, the instructor guides the program structure initially by example and continues by providing templates or program chunks as well as complete programs throughout the course. These examples of expert programming, coupled with careful monitoring of code listings, will give the students the freedom to develop their own logically sound methods of problem-solving. Coding skills are strengthened through practice. Instead of teaching a few commands and writing short programs to use them and then moving on to new commands, the cumulative design of this course demands the repeated use of nearly all commands from the time they are introduced through the end.

The fourth step, debugging, is often the most difficult part of the problem-solving process. The students must be able to figure out what happened as opposed to what they intended to happen to correct the error. To do this, they need to understand exactly what each command does to be able to follow the flow of control to figure out where the problem occurred and make suitable modifications. This skill should be enhanced by having the students regularly debug faulty programs supplied by the instructor.

An important ingredient of problem-solving techniques is the top-down approach which is breaking down a large problem into combinations of small problems. Each part is then solved so that when combined they form the solution to the large problem. The course accommodates this approach by its cumulative design. The course in general, as well as each unit, begins with simple problems and proceeds to increasingly complex programs utilizing all previous commands and problem solutions.

Each unit is divided into sub-units that present one or two new concepts and related commands. The problem(s) as the end of each sub-unit can be used or, if desired, any other suitable problem that exercises the current concept can be substituted at the instructor's discretion. If it is determined that the students need more practice to acquire a working template of a particular concept, more problems should be solved before continuing.

Because the focus of this curriculum is to develop thinking skills, rather than simple knowledge of programming commands, the students should be encouraged to use the included problems as a basis for the development of their own problems. This allows a tremendous amount of flexibility and serves to motivate and hold interest. The freedom of this approach also lends itself to the customization of the course to students of differing abilities. The more advanced students can incorporate more concepts, employ more complicated programming templates and/or solve additional problems. The opportunities for problem-solving growth are limited only by time and the student's ability.

Tests are suggested at the end of and occasionally during each unit. Before discussing the several items that should be included to assess the student's knowledge and thinking skills, the one thing tests should NOT be is a simple test of BASIC commands. This is merely rote recall and gives no indication of the problem-solving skills they should be developing. A two-part exam, given on consecutive days (one day of written work and one day of on-line work), is probably the best way to adequately cover the following types of assessments.

Written questions should be included to test the problem definition and planning procedures utilized by the students: Present a problem and have the student define the problem specifically and write an outline or set of algorithms to solve it. The knowledge a student has of the commands will be evident in the coding part of the test and the explicit understanding the students have of the commands should be tested by predicting the outcome of and debugging supplied programs. Pass out appropriate programs or program segments and require students to determine exactly how the program works and show the outcome. The debugging task is most effectively done on the computer. Supply a disk with buggy programs and have the students correct the bugs to render the program functional.

Student generated code should be evaluated for logical flow, modular programming and should include REM statements for each part of the program and all subroutines to make the program easy to read by others. The programs should also have a subroutine for any routine that is used more than once. Finally, to encourage modular, logical flow, the GOTO command should be used only as part of a conditional statement.

The course is divided into nine units. Each unit has a suggested length in weeks with the total equaling approximately 30. Since this curriculum was developed to be a problem-solving course, the first six weeks can be spent using problem solving software as a preliminary unit to the programming segment. If problem solving software is not available or desired, the BASIC course can easily be expanded to 36 weeks by requiring more programming problems and/or more complex problems as the course progresses.

Integrating Software into a Problem Solving Course

As concluded in the review of research contained in the first portion of this project, the computer is an effective tool for developing problem-solving and higher-level thinking skills. There are two major ways to accomplish this: Computer programming and software designed to enhance these skills.

The ideas presented here are intended to comprise the first four to six weeks of a problem-solving course based on computer programming. Software is used initially in the course to familiarize the students with the computer, thus reducing performance inhibiting anxiety and to raise their awareness of the process involved in solving problems. The four generally accepted steps are: (a) analyze, (b) plan, (c) try, and (d) check. These four steps are essentially the same as the four phases of programming/problem-solving—(a) specification, (b) planning, (c) coding, and (d) debugging—and should be used in the software section of the course and the programming segment.

The first two weeks are to be spent familiarizing the students with the computers and the general process of solving problems. Games are a good way to overcome the initial anxieties of using the computer, learning the ins and outs of the machine, and laying a foundation for the development of problem-solving skills. There are many games that are not oriented toward subjects normally taught in high school, but are fun to use and do require problem-solving skills. There are also programs designed to develop problem-solving skills in a game type or make believe situation. Try to have a variety of programs available and let the students choose which ones to use. Do not let them "cheat" by getting the answers or clues from other students. Encourage note taking and careful thinking to solve the games rather than randomly trying solutions. Once a program's problem has been solved, students should begin a new program rather than repeating the same one.

Some appropriate games:

Where in the World is Carmen San Diego?	Broderbund
Car Builder	Weekly Reader Family Software
Mystery Mazes	Educational Activities, Inc.
Heart of Africa	Electronic Arts

Specific problem-solving software in a game type format:

Incredible Laboratory	Sunburst Communications
Tribbles	Conduit
Discovery Lab	MECC
Discovery: Experiences in Scientific Reasoning	Millikan Publishing C0.

After the games have been used for one to two weeks, lead a class discussion about the four steps of solving problems. Discuss the games and relate them to the mental processes engaged by the students to work out their solutions. Talk about applying the processes to various games and problems presented by other school subjects such as math, science or English. Establish clearly the progression needed to be followed for all problems.

The next segment of the course will be similar to the first, except that the software will be oriented toward a subject normally taught in high school. The purpose is to apply the same problem-solving strategies to problems that are more relevant to the student. It will also help them see how to apply the strategies to different situations. Once again, it is important to let the students choose the first piece of software. This gives them a chance to develop the practical application of problem-solving in a field they enjoy. However, make sure successive programs are based on different content to help the student see how the same processes apply to other areas.

Some appropriate content oriented programs:

Math:	
Graphing Equations (Green Globs)	Sunburst
SemCalc	Sunburst
The Budget Simulator	EMC Publishing
Applying Mathematics	Laidlaw
The Whatsit Corporation	Sunburst
Survival Math	Sunburst
Social Studies and Geography:	
Social Studies Explorer Series:	Mindscape
Revolution and Constitution	
Westward Expansion	
Civil War and Reconstruction	
Central United States	
Western Europe	
Where in the World is Carmen San Diego?	Broderbund
Reading:	
Mystery Mazes	Educational Activities
Science:	
Botanical Gardens	Sunburst Communications
Oh, Deer!	MECC

The following part of the software unit is optional. If an intermediate step is desired between using applications software and computer programming, a logical choice would be the use of an authoring language or similar program requiring conceptual and procedural steps that are almost identical to those used in a programming language. Since most schools are limited to Apple II series computers, compatible authoring languages presently available are limited to Super Pilot. Authoring languages are simplified programming languages that are easy to use without requiring extensive time allotted to learning the program. If a program similar to HyperCard for the Macintosh becomes available for the Apple II , it would be an ideal choice. Authoring languages can help students learn concepts of program conditionals, structure, and program construction. The students can construct programs or games that they are interested in, as well as program problems presented by the instructor.

References

Burton, J.K., & Magliaro, S., (1988). Computer programming and generalized problem-solving skills: In search of direction. *Computers in the Schools, 4*(3/4), 63-90.

Clement, C.A., Kurland, M.D., Mawby, R. & Pea, R.D., (1986). Mapping the cognitive demands of learning to program, in R.D. Pea & K. Sheingold (eds.), Mirrors of Minds, 103-127.

Clement, C.A., Kurland, M.D., Mawby, R. & Pea, R.D., (1986). Analogical reasoning and computer programming, Journal of Educational Computing Research, 2(4), 473-485

Dalbey, J., & Linn, M.C. (1986). Cognitive consequences of programming: Augmentations to BASIC instruction. Journal of Educational Computing Research, 2(1), 75-93.

Dalbey, J., & Linn, M.C. (1985). The demands and requirements of computer programming: A literature review. Journal of Educational Computing Research, 1(3), 253-274.

Dalbey, J., Tourniaire, F., & Linn, M.C. (1986). Making programming instruction cognitively demanding: An intervention study. Journal of Research in Science Teaching, 23(5),427-436.

Kurland, D.M., Pea, R.D., Clement, C. & Mawby, R. (1986). A study of the development of programming and thinking skills in high school students. Journal of Educational Computing Research, 2(4), 429-455.

Magliaro, S.B.,& Burton, J.K. (1988) Adolescents' chunking of computer programs. Computers in the Schools, 4(3/4), 129-138.

McCoy, L.P., & Orey, M.P. (1988). Computer programming and general problem solving by secondary students. Computers in the Schools, 4(3/4), 151-157.

Palumbo, D.B. & Reed, W.M. (1988). Intensity of treatment and its relationship to programming problem solving. Computers in the Schools, 4(3/4).

Peterson, C.G. & Howe, T. G. (1979), Predicting academic success in introduction to computers. AEDS Journal, 12, 182-191.

Reed, W.M. & Palumbo, D.B. (1988) The effect of the BASIC programming language on problem-solving skills and computer anxiety. Computers in the Schools, 4(3/4).

Reed, W.M., Palumbo, D.B. & Stolar, A.A. (1988). The comparative effects of BASIC and logo instruction on problem-solving skills. Computers in the Schools, 4(3/4).

Soloway, E., Lochhead, J. & Clement, J. (1982). Does computer programming enhance problem-solving ability? Some positive evidence on algebra word problems. In R.J. Seidel, R.E. Anderson & S.B. Hunter (eds.), Computer literacy: Issues and Directions for 1985 (pp.171-185).

Webb, N.M. (1985). Cognitive requirements of learning computer programming in group and individual settings. AEDS Journal, Spring 1985, 183-194.

Appendix: Sample Programming Unit

UNIT 4: TEXT COMMANDS FOR SCREEN FORMATTING AND AN INTRODUCTION TO STRING VARIABLES

Unit length 2 Weeks

UNIT OBJECTIVES: The students will-

- learn some text formatting commands.
- use a delay loop
- use simple text commands to introduce a graphics picture and embedded DOS commands to automatically run it.

TERMS: Embedded DOS commands, Text mode vs Graphics mode, Text formatting, Delay loop

COMMANDS: TEXT, FLASH, INVERSE, NORMAL, SPEED=, HTAB

VTAB, : colon for linking lines together , FOR x = 1 to 500 : NEXT x, PRINT:PRINT CHR$(4); RUN "xx", PRINT:PRINT CHR$4(4); BLOADA$2000

MATERIALS: List of commands needed and their function,
Sample program to show and pass out.

SUB-UNIT 1 Sub-unit length: 3 Days

OBJECTIVE: The students will-

- use the SPEED=, HTAB, VTAB, INVERSE, FLASH and NORMAL text formatting commands to create an attractive paragraph.

PLAN: Point out that, although graphics may be used to accompany future projects, graphics programming instruction is complete. The task at hand will focus on some text commands which will be used to manipulate the appearance of text on the screen display. The print statement was used before to "title" some Lo-Res graphics, but the computer has never been used in the text mode. Explain that this unit will consist of: 1. an introductory paragraph will be written for either a Hi-Res or Lo-Res graphics program that has already been created, 2. a keypress will trigger the picture display, and finally, 3. a delay loop will cancel the picture and cause another message to be displayed.

The first step is to do some text formatting. Point out that one of the first steps done in graphics program was to issue a GR, HGR command to invoke the graphics mode. The computer is automatically placed in the text mode, which is why the graphics call had to be made. Now the text mode will be used to present text, a paragraph, explaining a previously drawn graphics picture. Give the screen size for text (40 characters by 24 lines). Discuss the appearance of text making the following points:

- Do not fill an entire screen with text-it is hard to read.
- It is easier to read double spaced text than single.
- It is often nice to leave some space at the top of the screen.
- It is often nice to leave space at the left side of screen.

- Attention can be drawn to words by changing their appearance.

INVERSE-creates a white background with black letters.

FLASH-alternates between the inverse and normal text.

A PRINT statement without any text will print a blank line.

VTAB (y) moves the cursor to the specified line

HTAB (x) moves the cursor to the specified column. (Any of these statements can be combined to control the text)

INVERSE invokes the inverse mode

FLASH invokes the flash mode

NORMAL sets the mode back to normal.

SPEED= alters the presentation of successive characters on the screen. The default value is 255. Make sure SPEED is set back to 255 at the end of a program.

Mention that program lines can be combined with a colon.

If a sentence has NORMAL and FLASH/INVERSE text, the semi-colon is used to keep it all on one line:

```
10 PRINT "normal text";FLASH; PRINT "flashing text"; NORMAL; PRINT"normal text again"
```

At this point involve the class in creating the beginning of a program to display text.

PROBLEM:

HOMEWORK: a paragraph of about 100 words to describe a picture of choice. Decide what the screen display will look like and mark the paragraph. Include margins to utilize HTAB and VTAB, and also FLASH and INVERSE. The code to present the paragraphs do not have to be coded.

INCLASS: The paragraph written for homework will be coded in class the next 2-3 days. The students should have a sample program to refer to.

SUB-UNIT 2 Sub-unit length: 2-3 Days

OBJECTIVES: The students will-

- use FOR/NEXT loops to delay text presentation.
- use an embedded DOS command to run a program.

TERMS: Delay loop, Variable, Embedded DOS commands

COMMANDS: FOR/NEXT, PRINT:PRINT CHR$(4); "RUN xx"

PLAN: The class has written a paragraph to introduce a graphics program. Explain that since their introductory paragraph is a different program from the graphics program, there must be a way of starting the graphics program after the text program without going through LOAD xxxx and RUN xxxx. They will be familiar with the term "DOS command" by now, so merely tell them that a DOS command may be placed in a program so it can be executed"automatically." Put an example of the RUN "xx" command on the board and point out that it is necessary to let the computer know that a DOS command is about to follow by prefacing the command with PRINT:PRINT CHR$(4);. If the programmer wishes to run another program after the completion of one program the following command is necessary assuming the last line of the program is line 390:

```
390 PRINT:PRINT CHR$(4); "RUN xx"
```

When the first program is complete and line 390 is encountered BASIC will automatically RUN xx.

CAUTION: Make sure to stress the importance of SAVING a program that contains the embedded DOS command RUN before running it. When a new program is run, the old program is purged from memory. If there were any changes made to the program before it was run, they will be lost.

Before the students add the embedded RUN command to their programs, it is necessary to explain Delay Loops. Begin by explaining that their introductory paragraphs will be seen only briefly because the computer will print the text and then automatically go to the RUN command. The computer can be slowed down to give the reader a chance to finish the text by making the computer count. The higher it has to count, the longer the delay. To make the computer count we need to add some lines after the introductory text and before the RUN command. Explain that a subroutine containing a counting loop is to be created in which the computer stays and counts until it reaches the amount determined by the user. The loop begins by naming a variable and specifying the range. Write on the board:

```
1000 REM ******* Counting routine ********
1010 FOR COUNT = 1 TO 500
1020 NEXT COUNT
1030 RETURN
```

Explain the use of the variable and the program control, how it starts at 1, adds to itself when it reaches NEXT, goes back to FOR etc. until it reaches 500. Emphasize that the upper limit should be varied until the loop is an appropriate length. Now insert the FOR/NEXT loop in the hypothetical program on the board between lines 380 and 390 by using the

GOSUB command to send it to the counting subroutine.

PROBLEMS: Add a subroutine to the text programs to delay the presentation of the second page of text if you have one. Add the the embedded RUN command.

BONUS: Use a delay loop to delay certain parts of the text. If there are different length waits desired more subroutines must be created.

EVALUATION: Programs will be printed and turned in for a grade. The show and tell will be graded.

SUB-UNIT 3 Sub-unit length: 3 Days

OBJECTIVE: The students will-

- gain a rudimentary understanding of the difference between a numeric variable and a string variable.
- understand and use the GET command to create a "page turn" subroutine.
- understand that one subroutine may be called from another.

TERMS Numeric variable, String variable

COMMANDS: GET x$

PLAN: Begin by recapping all the modifications made to the animation program. Call attention to the fact that there are many different reading speeds as evidenced by the differences in the delay loops presented at the show and tell. The point of this class will be to make additional changes to allow the page to be turned when the READER wants. To do this a subroutine will be created that waits for the reader to press a key on the keyboard when he/she wishes to continue. Lead a discussion to create the subroutine on the board:

```
5000 REM*****Page Turn *****
5010 VTAB 24
5020 PRINT "Press any key to continue"
5030 GET A$
5040 HOME
5050 RETURN
```

Naturally, the explanation about String and Numeric variables will take place when line 5030 is encountered. Point out that previously the variable was numeric because the computer performed a math function on it. The word string means any alphanumeric character is to be used for something but not a math function. When the GET command is encountered, the computer waits for any alphanumeric character to be typed in by the user. The character is stored under the variable name, just like the numeric variable was, but in this case we are not going to use it for anything but a signal for the computer to continue executing.NOTE: This a brief discusion of string and numeric variables-more detail is contained in the next unit.

Before the discussion ends, draw attention to the fact that the delay loops should still be used to delay the presentation of the page turn prompt. The length should be shortened to an appropriate length. Demonstrate the program structure on the board incuding a delay subroutine and a page turn subroutine. Go through the program control.

PROBLEM: Make the necessary changes to the text-graphics program to do the following:

The intro program should be split into two pages if not done already
Use appropriate margins and text placement
Incorporate the page turn routines
Write a short program to add text after the graphics

Insert another embedded DOS command in the graphics program to run the closing program. OPTIONAL: Add some text to the text window in the graphics mode. Point out that if Hi-Res graphics is used, a VTAB command is necessary to place the text in the text window.

EVALUATION: The show and tell will be graded. The program listings will be graded. Compare to the previous edition.

TEST: Create a one day test that includes:

Programming with the text formatting commands
Program prediction
Program debugging

Paper

The Need for Education *and* Technology: Examples from the GPCeditor*

Mark Guzdial
The University of Michigan

Abstract

The GPCeditor (Goal-Plan-Code editor) is a computer based environment for learning software design in high school. Case studies suggest that students using the GPCeditor do make use of its features to learn software design. However, developing solutions to the problems that the students still face requires addressing both technological and educational issues. What has emerged is an awareness of the interrelationship between instruction, curriculum, and technology.

I. Introduction

A problem in American schools is a lack of learning about design skills. Students are taught to analyze problems (e.g., write book reports and solve story problems), but they are not taught the skills of synthesis or design. These skills include identifying alternative solutions and assembling components of a solution (Simon, 1969).

To address this issue, we of the HiCE Research Group[1] constructed a computer based environment for learning software design skills, and used this technology in an Ann Arbor high school. Software is a particularly good domain for studying design since a well-defined design is in fact an executable program. The GPCeditor (Goal-Plan-Code editor) helps students learn design by presenting a structure within which to think about programs, and a process which has proven useful for expert designers.

After studying the students use of the GPCeditor, the complex interaction between educational issues (especially, curricular and instructional issues) with technology became more obvious. This paper describes the instructional package that we placed in the classroom (GPCeditor, curriculum, and instruction) and the students use of it. Section two covers the features of the GPCeditor in terms of the structure and process supported, the instructional environment that evolved, and the curriculum used in the classroom. Section three contains case studies of two students selected as a representative sampling to identify benefits and problems with our approach. Finally, section four summarizes the benefits of this approach and the directions that we plan to follow in addressing the problems identified.

II. Components of Instruction

A. The GPCeditor

The GPCeditor (Goal-Plan-Code editor) is a computer based environment in which students can learn software design in Pascal. It runs on Apple Macintosh computers with 19-inch monitors[2] and at least 2.5 Mb of memory. Through a donation from Apple Computer, eleven stations were placed in Community High School in Ann Arbor.

Supported Structure: The structure through which students develop programs in the GPCeditor has two components. The first is the goal-plan approach, an expert-like way of thinking about programs as other than lines of code. The second component is the multiple representations used to support this way of thinking.

Students using the GPCeditor never type program statements. Instead, they construct programs by defining goals (statements of what they wish to achieve) and plans (how they wish to achieve these goals). For each goal, there are one or more plans for achieving the goal. These plans can be thought of as components, or program modules, which are composed by the student to create the complete design or program. These plans are either defined in terms of Pascal code (e.g., a **writeln** statement is a primitive plan available to the student) or in terms of a hierarchy of subgoals and their plans. The role of data is to instantiate a plan for a particular program. For example, the **writeln** plan is always the same from use to use and program to program, but it is made specific for a particular purpose by the data that it writes.

For example, we hope that a student using the GPCeditor, when asked to write a program to add two input numbers together until the user indicated that he was done, would not address the problem by writing a program with **readln**, **writeln**, and a **while-do** loop. Instead, we hope that she thinks about a **GetInput** plan (consisting of a **writeln** to write a prompt for the user and a **readln** for reading a value) which asks users if they'd like to begin adding numbers and a **DoItAgain** plan (containing a **while-do**, a test for equality, and a **GetInput** plan at the bottom of the loop) which performs some plan and uses a **GetInput** to ask users if they would like to do the plan again. The student might also think about using a couple of **GetInput** plans for reading in the numbers.

---

* This research was supported by NSF Grant #MDR-9010362.

[1] The Highly-Interactive Computing Environments (HiCE) Research Group at the University of Michigan, directed by Elliot Soloway. The GPCeditor group include Kathy Brade, Luke Hohmann, Iris Tabak, and School of Education Professor Phyllis Blumenfeld.

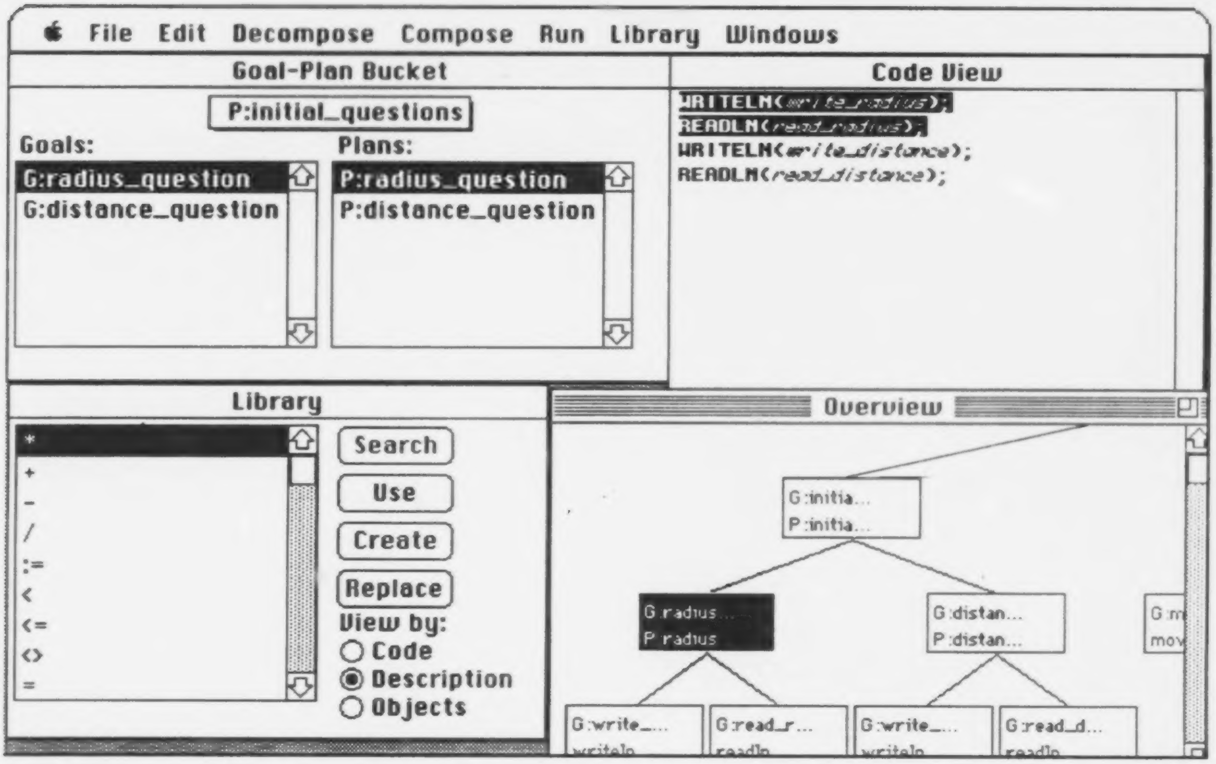
[2] A new version of the GPCeditor runs on 13-inch monitors.

Research on expert software designers suggests that they use such a structure when thinking about programs (e.g., Soloway & Ehrlich, 1984). By thinking about the design in terms of what's to be accomplished and plans that meet those needs, the designer changes the design task from thinking about code and syntax to thinking about higher-level components. The design skill of defining a task in terms of goals and plans is *modular decomposition*. It has been emphasized by design researchers as key to coping with design complexity (e.g., Parnas, 1972). Thus, the goal-plan structure enables students to think about programs using the same modular decomposition techniques that experts use, without having to learn about scoping and parameters in order to use the traditional module constructs (procedures).

A goal-plan approach reduces design complexity by fragmenting the task and providing a framework for component reuse. A plan defined in terms of subgoals defines a level of hierarchy in the program which can be dealt with almost as a program unto itself. Thus, a program in the GPCeditor is not a monolith, but a collection of small, manageable pieces. Experts reuse pieces such as these and attempt to solve new problems using old, tested pieces. This reuse of plans further reduces the complexity of design by allowing the designer to treat entire branches of the decomposed task as a solved problem. The GPCeditor supports reuse with a Plan Library (lower left corner of Figure 1). The student's library initially contains plans for all of the standard Pascal components, based on those included in traditional Macintosh Pascal implementations. Students can add their own plans to the library (such as **GetInput** and **DoItAgain**) for later reuse.

The GPCeditor provides multiple, linked representations to support students using a goal-plan approach to programming. These representations make it possible to manipulate, concretize, and think about the goal-plan structure. We provide more than one representation because use of multiple representations is another expert-like technique that we seek to foster with the GPCeditor. Researchers have found that studying a problem from multiple representations can make a solution more evident than might be possible from any one representation (Larkin & Simon, 1987).

Figure 1 is an example screen from the GPCeditor. The upper left window is the Goal-Plan Bucket which contains goal-plan lists. These identify, by name, each goal and its corresponding plan at a particular level of hierarchy (named above the lists). In the example, the named goals and plans implement the **initial_questions** plan. The upper right window presents the traditional code-oriented view of the program. The lower right window gives a graphical Overview of the hierarchy of


File
Edit
Decompose
Compose
Run
Library
Windows
Goal-Plan Bucket
P:initial_questions
Goals:
G:radius_question
G:distance_question
Plans:
P:radius_question
P:distance_question
Code View
Library
*
+
-
/
:=
<
<=
<>
=
Search
Use
Create
Replace
View by:
○ Code
● Description
○ Objects
Overview
G:initia...
P:initia...
G:radius...
P:radius
G:distan...
P:distan...
G:write_...
G:read_r...
G:write_...
G:read_d...

```
WRITELN(write_radius);
READLN(read_radius);
WRITELN(write_distance);
READLN(read_distance);
```

Figure 1
GPCeditor Example Screen

goals and plans. Each goal appears with its corresponding plan, and if the plan is defined in terms of lower level goals and plans, those appear below and connected to the goal-plan node.

These representations are linked such that corresponding elements in all the representations appear highlighted at the same time. Clicking on a goal name in the goal list highlights the corresponding plan in the plan list, the corresponding code in the Code View, and the corresponding node in the Overview representation. The purpose for the linking is to encourage students to make the transition between working with one representation to working with several.

Supported Process: The structuring features of the GPCeditor provide students with a way to think about programs as experts do. The process features of the GPCeditor allow students to use these structures as experts do.

The design model that we seek to support using the GPCeditor is based on other models of student and expert programmers (Spohrer, 1989; Adelson & Soloway, 1984). The model has four main stages, with subprocesses beneath each stage. These stages are:

- **Reflection** during which the student reflects on the problems, decomposes all or part of it, and formulates a goal to achieve. An expert's instance of the entire design process is marked by frequent reflection on the task and the process (Jefferies, Turner, Polson, & Atwood, 1981).
- **Planning** during which the student considers plans that meet the goal and chooses one of these to implement. The student then instantiates the plan for the particular program by identifying data objects for the plan.
- **Generating and Evaluating Alternatives** during which the plans are ordered and composed to generate a final program. An expert programmer will also review and evaluate the program at this point, perhaps generating alternative decompositions or compositions.
- **Debugging**, which is actually a cycling of the design process through Reflection and Planning.

The student steps through this process using items in the **Decompose** and **Compose** menus. For example, the student uses **New Goal...** under the **Decompose** menu to formulate a goal to be achieved. **New Plan...** in the same menu is used to choose a plan for the goal. When a plan has been defined, the student must instantiate the plan for the particular purpose using data objects in the program in a representation called the Match Window. Once instantiated, the plan can be ordered and composed into the program using the tools in the **Compose** menu. Finally, the program can be executed. When executed, program output appears in the Text and Graphics windows, while the Observe window describes the program's data objects and their current values.

B. The Instructional Environment

The high school in which the GPCeditor is used is Community High School, an alternative school in the Ann Arbor school district. The high school is loosely-structured, emphasizing individual creativity and inter-disciplinary efforts. Students don't have homerooms: they have forums that might as likely meet in the evening at an area theater as in a formal classroom. Students are neither necessarily gifted nor learning-disabled. They come to this school because they are uncomfortable with more traditional schools.

Students apply to enter the course. Selection of students for the course is made by the administration, the course instructor, and members of the HiCE research group. The goal is to fill the class with a wide range of students on dimensions such as gender, race, past computer experience, and academic performance. Ten to eleven students take the class each semester. The class using the GPCeditor has been run every semester since Winter 1989. The students discussed in the next section used the GPCeditor in Fall 1989.

During the semester I studied the students, the class met five days a week for two hours a day[3]. Since a two hour class was unusual for Community High, the class was scheduled for the last hour of the day and for an hour after school as well.

The instructor for the course has been teaching mathematics (algebra, geometry, and calculus) for more than ten years, and computer science (specifically, Pascal) for seven years. He, with members of the HiCE group[4], evolved the style of classroom interaction during the first semester of the course. Each student progresses through a set of worksheets at his or her own pace. The instructors wander the classroom, providing individual instruction when asked, and offering advice or asking students about their projects to encourage reflection. Occasionally, the class gathers to discuss a common topic. For example, a common group activity is to discuss what makes a good design. The instructor presents student's programs in their Overview representation on an overhead projector and has the students review and critique each design. Is the depth of decomposition appropriate? Is the branching reasonable? Would a different decomposition have been better?

After approximately twelve weeks of the sixteen week semester, some students begin using Lightspeed Pascal instead of the GPCeditor. Most of the students (50 to 75%) finish the portions of the curriculum that can be completed on the GPCeditor. They then use a traditional Pascal environment for features such as procedures and arrays which were (at the time of the study) not yet implemented in the GPCeditor[5].

C. The Curriculum

The curriculum used in the GPCeditor class was originally developed by Jeannine Pinto and Yasmin Kafai at Yale University for use in a course at a New Haven high school. The New

[3] The class now meets three days a week for ninety minutes, with after school sessions for those students who want additional help.

[4] During the first month or two of the first semester using the GPCeditor, three instructors were present daily (the class instructor, Soloway, and one of his research assistants, Yasmin Kafai). Once a style of interaction was established, the number of instructors dropped to two (the class instructor and a research assistant).

[5] Procedures have since been implemented into the GPCeditor.

Haven course was an attempt by Soloway and his group to teach goal-plan structuring and some of these other design skills using a traditional Pascal programming environment. Without support for the concepts in the environment, they found that students continued to focus on the lines of code. This curriculum was updated by Kafai for use with the GPCeditor.

The course is self-paced. Students complete worksheets that discuss topics in Pascal and design. They file their completed worksheets and programs in individual notebooks, which the instructor collects, comments on, and grades every two weeks. Table 1 lists the GPCeditor worksheets. Each worksheet has a particular focus, as indicated by the table. Some emphasize one or more Pascal constructs, some emphasize phases of the software design process, and a few focus on use of the GPCeditor. There are 27 worksheets and 41 assignments in the semester.

The goal of the curriculum is to teach students to develop Pascal software using good design skills in the expert-based software design process described earlier. The curriculum is organized towards developing increasingly complex programs throughout the semester while simultaneously learning the design skills to cope with this complexity. As students learn more Pascal and are asked to integrate and use their Pascal knowledge as they create complex programs, they are introduced to design skills to make the design of their program easier. Worksheets that discuss issues such as identifying plans and where decomposition is useful are interwoven with worksheets on **while-do** loops and Macintosh graphics primitives.

III. Case Studies

Two case study students are described here. These students were selected as a representative sample of the types of student use of the GPCeditor. The data used to describe these students comes from five sources.

- **Think-aloud protocols**: At three points during the semester, students were asked to complete design tasks while thinking-aloud. Video and audio recordings of the students were made. The protocol tasks were different from those in the worksheets, but required similar skills. The tasks consisted of three to four programs organized such that plans from the first programs could be reused in later programs. For example, the first protocol asked students to draw a face on the screen, then to draw two faces, then to draw two faces and display a greeting to the user.
- **Student journals:** Students were asked to keep daily journals as designer notebooks (as in Harel, 1988). They were asked to address what they did each day, the problems that they had, and what they planned to do next.
- **Interviews:** Students were videotaped in interviews with a researcher at the beginning of the semester (second day of class). They discussed why they were taking the class, whether they had programmed before, and whether they liked computers.
- **Usage trace files:** The GPCeditor automatically collects records of student actions while in the GPCeditor. These recordings provide data on how the student used the GPCeditor and what features were used.
- **Class programs**: The students' programs written to meet worksheet assignments were also collected with the trace files.

A. Allen

Allen[6] took the GPCeditor class because he was interested in computers. He had studied Basic for six weeks when he was in sixth grade. He took the class as a sophomore with a 3.0 GPA.

Allen was immediately successful with the GPCeditor. By the end of the first week, he was almost a week ahead of the rest of the class in terms of worksheets and assignments completed. His first protocol, after three weeks of GPCeditor use, indicated an already advanced grasp of design skills.

- He constructed a hierarchy of goals and plans, and he understood how to use the hierarchy.

> "Let's add a little hierarchy here..."
>
> *Why did you create that?*[7]
>
> "So I can put all the circles, the face part in that. And then if I put the face part in that, then I can just...add the different parts of the program, add those in a different part."

- He reflected on the task and made explicit plans.

> "How do we do this? Get their name, then draw the faces, and finally display hello and their name."
>
> "The vertical would be...Let's a draw a picture for a second."

- He used the various representations when reflecting on the program.

> *(While clicking in the Overview window)* "I always like to see what's going on over here, just to see where I am, what it looks like."

However, Allen's class programs were not always models of good hierarchical decomposition. In his second protocol session, programs that he began with a comment like "this is easy" invariably had little or no hierarchy (i.e., he built his program entirely out of Pascal primitives as opposed to defining levels of hierarchy. However, when he encountered a program bug, for example, he would often define a new level of hierarchy and decompose the difficult plan there. For Allen, hierarchy seemed to be a method of coping with complexity.

---

[6] Student names have been changed to protect subject anonymity.

[7] Verbal comments by the researcher appear in italics. Comments in parentheses explain activity during the session.

Allen rarely saved his plans to the plan library. When asked why in class one day, he replied that it was "easier to build it up." Over the course of the entire semester, Allen's usage trace files show that he saved only one plan to the library.

By the end of the semester, Allen's process was expert-like. He made few mistakes in his third protocol session. His usage trace files showed few program modifications and few program executions. As an example, during the first four weeks of the semester, Allen ran his programs an average of 14 times a day. During the last four weeks (when he still used the GPCeditor before going on to Lightspeed), he ran his programs only 3 times a day. His design process seemed to have been learned and internalized.

B. Sue

Sue entered the GPCeditor class as a junior with a 3.7 GPA. She had done some programming in Basic. In her initial interview, she expressed frustration in her experience with computers:

> *Do you like computers?*
>
> "If I know what I'm doing, I like them. If I don't understand, I get frustrated."
>
> *Is that (frustration) the hardest part of using computers?*
>
> "Yeah, just, in simple terms, I don't know, trying to express the ideas that you want, and trying to get it on to the screen, trying to figure out what steps you have to do to get things on to the computer or whatever."

Sue had difficulty programming from the start. She took longer than the other students and required extensive help. On a typical program, Sue took over three times as long as other students, ten days versus three days, and she mentioned in her journal receiving help on four of those ten days.

Part of the reason that she took so much longer writing her programs was that she took great pains to have good hierarchical decomposition and plan reuse in her program, going so far as to redesign programs in order to achieve these goals. The usage trace files show that she saved ten plans to the library during the course of the semester–more than twice as many as any other student in the class. The following quote from her journal exemplifies the effort that she went through for good decomposition and plan reuse.

> "I retyped the Challenge so that I could have Hierarchy and reuse some of the same plans and goals. It really wasn't that hard because I knew what I was doing but now I have to change things so that it will make sense for the next loop in the programming."

Sue's first protocol session was marked by a lack of planning. She did not reflect on the task and her process early, but late when it was difficult to backtrack, for example, while instantiating a plan for a particular program. In particular, she lost track of data objects and her rationale for the changes she was making. She created no hierarchical decomposition.

- She had difficulty remembering her data objects and her debugging process.

> "Oh? What did I do? Oh, I'll have to change that. The vertical. I forgot what I changed."

- She spent a good deal of time staring at her code and then made changes almost randomly.

> (Sue stares at the Code View , clicking on various pieces)
>
> "Hmm, I'm not sure..."
>
> (She clicks on a plan to draw a circle, moves it one statement further down in the program — a change that has no effect on execution. She runs the program and sees no difference. She clicks on the Code View again.)
>
> "I had it right before."
>
> (Clicks on the output graphics window.)
>
> "Maybe it's here."
>
> (She changes the value of a data object, then re-runs the program.)
>
> "Okay, that did make a difference."

- She never referenced any representation other than the Code View.

Her second protocol showed little improvement. For example, she uses hierarchy, but only in a strictly rote fashion, and she still has difficulty tracking her data objects and strategies.

- She still has difficulty remembering her data objects and remembering why and where she makes changes in her program.

> "I don't know—I'm just trying to figure out. Can I just match this? *(Pointing with her mouse to a data object.)* Left and right would be horizontal, up and down would be vertical? What should I do? I don't know."

- She announces that she's using "Plenty of hierarchy." However, she only decomposed **GetInput** plans and bodies of **while-do** loops, two purposes explicitly suggested in the worksheets. She never used a new level of hierarchy, for example, to identify a logical partition in the program.
- In one protocol task, she was asked to write the result of a calculation. She created a new level of hierarchy, placed a **writeln** within it to write the result, then placed a **readln** immediately afterward — as if she was creating a **GetInput** plan.

Table 1
GPCeditor Worksheets and their Focii

Worksheet	Focus
Planning, Programming, and Problem Solving	Introduction to the GPCeditor. Students are asked to compose existing library plans and execute their programs.
Values and Variables: Viva La Differences	Students learn the difference between values and variables. They construct simple programs for performing math operations.
User Friendliness: The Key to Popular Programs	Introduction to the **GetInput** plan (prompt the user for data entry, and input of data).
Getting Coordinated: An Introduction to the GPCeditor Coordinate System	Students learn to specify locations in the Graphics window in terms of horizontal and vertical offsets.
Graphics in the GPCeditor	Introduction to Macintosh graphics primitives, such as **paintoval** and **framerect**.
Looping in the GPCeditor	Introduction to the **while-do** loop.
Moving Pictures in the GPCeditor.	Combining iteration with graphics to achieve animation.
Writing Good Programs	Identification of named plans such as **GetInput** and **DoItAgain**.
Plans for Moving Pictures	Descriptions of plans used in animation.
Problem Simplification	Discussion on program decomposition.
The Word of the Day: Hierarchy	Identifying program hierarchy.
Line Drawings	Introduction to **moveto** and **linteo** graphics primitives.
Blinking Eye Face	Draw a face with animated, blinking eyes–a complex program requiring 20-50 lines of program code.
Decisions, Decisions, Decisions	Introduction to the **if-then**.
Apply Your Knowledge: Identifying Plans	Identifying plans in other students' programs.
Finding out what you know about Plans	Quiz on plans.
Working with Plans	Introduction to debugging techniques.
Plan Library	List of all programming plans met in the course.
Random Numbers	Introduction to **random**
Random Numbers and Game Programs	Creation of computer games using **random** and **if-then**.
User-Friendly Interfaces	Using mouse-driven input.
Procedures	Introduction to procedures.
My Life in Disorganized Crime	Methods for organizing code (procedures).
OR: Logical Alternatives	Using logical conjunction.
Procedures revisited	More on procedures.
Handling User Input Errors	Handling data verification.
What you do when everyone wants to play: Arrays	Introduction to arrays.

- She did use the Goal-Plan Bucket to select plans and goals, but still did not reference the Overview.

IV. Conclusions

The features of the GPCeditor were used effectively by some students who made significant improvement in their design process. In particular, we can identify several design skills at which students did well and that the GPCeditor explicitly supported.

- Some students did seem to value and use concepts like hierarchical decomposition. This is significant since some researchers have argued that students find modular programming to be too great a cognitive burden (Pea & Kurland, 1986).
- Some students did make use of the plan library for reuse of modules (e.g., Sue), but not as much as we had hoped nor in the thoughtful way we had hoped.
- Some students did make use of alternative representations to aid in their reflection on the program.

A. Changes to the Instruction

Sue made some use of the facilities of the GPCeditor, but not with reflection on their purpose. In protocols, she used hierarchy and the library rarely and only in a rote manner. Her use of hierarchy and the library in her class assignments produced a good product, but only after considerable effort and with significant support. This seems to us to be a problem with not learning the process. She used the structure, but not in the way of an expert.

The solution to this problem may be modelling. As design processes are dynamic, they are difficult to transmit via worksheets, and thus require the instructor to model the design process (Collins, 1988.) By performing the task, using the tools, and explicitly demonstrating good process, the instructor can provide a model for the students to follow (Paris & Winograd, 1989).

B. Changes to the Curriculum

Though Allen developed significant design skills, we were surprised to note his lack of use of the library. However, upon review of the curriculum, we discovered that the curriculum only once asked students to save and reuse a plan (a **GetInput**) as part of an assignment. It is not surprising, then, that Allen did not reuse plans since he was given little guidance on how to reuse and when to do it.

We realized that our curriculum had been designed to teach Pascal and design skills, but without determining which skills and how they should be taught. Our coverage of the process was incomplete. We have since made an analysis to determine the design skills that we need to teach and are now updating the curriculum to cover those skills.

C. Changes to the GPCeditor

The GPCeditor can be changed to both provide better support to the students and to enhance the instruction and the curriculum.

- Prompts can be added to the GPCeditor to provide more explicit support for process learning. Example prompts might inform students when they should reflect or suggest what they might be thinking about for effective problem-solving (Polya, 1945).
- An explicit representation of the design process can be implemented as a tool for reflecting on the process. Making the goal of the instruction explicit has been shown to aid learning (Brown, Bransford, Ferrara, and Campione, 1983).
- New representations can be added to insure complete coverage of the design process. In particular, tools can be created to ease the process of tracking data objects and of recalling past debugging activity.

Any changes to the GPCeditor must be made flexibly, however, so as not to hinder the advanced students. Allen was successful in using the GPCeditor. Additions of these new features must be made without preventing advanced students from learning.

In conclusion, the GPCeditor has performed admirably in a classroom environment. It does provide the structure and process to encourage the learning of design skills. However, the next stage in its development requires attention not only to the lessons of technologists, but to the lessons of educators.

Acknowledgments

The GPCeditor was designed and mostly implemented by Ken Ewing at Yale University. A team of programmers completed implementation at the University of Michigan: Luke Hohmann, Dave Koziol, Dan O'Leary, Charles Weaver, and me. Luke Hohmann has maintained and updated the environment for the last two years. Thanks to Bob Kinel and Dean Bob Galardi of Community High School for their efforts in this project. Special thanks to Barbara Ericson, Stacie Hibino, Luke Hohmann, and Elliot Soloway for helpful comments on this paper.

References

Adelson, B. & Soloway, E. (1984). A cognitive model of software design. Technical Report #342, Cognition and Programming Project, Yale University, New Haven, CT.

Brown, A.L., Bransford, J.D., Ferrara, R.A., & Campione, J.C. (1983). Learning, remembering, and understanding. In Kessen, W. (Ed.), *Handbook of child psychology: Cognitive development*. New York, NY: Wiley.

Collins, A. (1988). Cognitive apprenticeship and instructional technology. Technical Report #6899, Bolt, Beranek, and Newman, Cambridge, MA.

Harel, I. (1988). *Software design for learning: Children's construction of meaning for fractions and Logo programming*. Unpublished doctoral dissertation, MIT, Media Technology Laboratory.

Jefferies, R., Turner, A.A., Polson, P.G., & Atwood, M.E. (1981). The processes involved in designing software. In Anderson, J.R. (Ed.), *Cognitive skills and their acquisition*. Hillsdale, NJ: Lawrence Erlbaum Associates.

Larkin, J.H. and Simon, H.A. (1987). Why a diagram is (sometimes) worth ten thousand words. *Cognitive science*, 11, 65-99.

Paris, S.G. and Winograd, P. (1989). How metacognition can promote academic learning and instruction. In Jones, B.F. and Idol, L. (Eds), *Dimensions of thinking and cognitive instruction*. Hillsdale, NJ: Lawrence Erlbaum Associates.

Parnas, D. (1972). On the criteria to be used in decomposing systems into modules. *Communications of the ACM, 15*(2). 1053-1058.

Pea, R.D. & Kurland, D.M. (1986). On the cognitive effects of learning computer programming. In Pea, R.D. & Sheingold, K. (Eds.), *Mirrors of minds*. Norwood, NJ: Ablex Publishing.

Polya, G. (1945). *How to solve it: A new aspect of mathematical method*. Princeton, New Jersey: Princeton University Press.

Simon, H.A. (1969). *The sciences of the artificial*. Cambridge, Mass: MIT Press.

Soloway, E. & Ehrlich, K. (1984). Empirical studies of programming knowledge. *IEEE Transactions on Software Engineering, 10*(5):595-609.

Spohrer, J.C. (1989). *MARCEL: A Generate-Test-and-Debug (GTD) impasse/repair model of student programmers*. Unpublished doctoral dissertation, Yale University. Also Technical Report YALEU/CSD/RR #687.

Projects
Language Arts Projects : K-University

The Computer as a Writing Classroom

Lance Coon
Phoenix Country Day School, Arizona

Abstract

This year, a new Macintosh computer network has become an integral part of the English curriculum at Phoenix Country Day School. Students not only use the lab on their own time to prepare all their English assignments, they also receive instruction in the features of Microsoft *Word* to help them outline, draft, revise, print, and store their written work. Most English classes incorporate regular visits to the lab into their weekly schedules, so that students receive both time and supervision as they work on their written assignments. Other features such as computer logs and correspondence journals augment the ways that the network can be used in the instruction of writing.

This project presentation documents how one teacher uses the network in his English composition and literature classes. The presentation itself consists of a verbal summary accompanied by a packet of handouts containing descriptions of procedures and illustrative samples of work produced using the network.

Cooperative Learning Strategies with Logo Writer

Sue Horn, Kaye Coats, and June Ehinger
Sand Springs Public Schools, Sand Springs, Oklahoma.

Abstract

National attention has been focused on restructuring classroom instruction. Departure from methods of drill and practice to those which empower students to direct their own learning are being suggested. Cooperative learning is an instructional strategy in which students are assigned to small groups to help each other learn. Interdependence is a goal. A project using Logo Writer was introduced to students in computer literacy classes in a cooperative learning format. This gave the students experience in developing individual and group goals, solving conflicts and problems in order to accomplish these goals and finally individual and group evaluation of their finished product. This presentation will describe the introduction of the project to the students, the formation of cooperative learning groups, problems encountered and solved, and lesson plans used. A brief video tape provides an opportunity to observe students in a cooperative learning environment sharing ideas and solving conflicts as they develop their Logo Writer projects.

Researcher & Programmer: A Collaborative Approach to Developing Research-Based Software

Janet Graeber and Roger Knight
University of San Francisco

Abstract

During the past decade many teachers have incorporated instructional and tool-based software as an integral component of the curriculum. With the availability of HyperTalk-based programming languages for microcomputers, software can now be created to empower the classroom teacher as researcher. Both ease of use and natural fit within the curriculum is a major consideration. In an attempt to explore this challenge, the authors co-developed a writing research tool that uses the Macintosh and HyperCard.

During the software development process, the researcher assumes primary responsibility for the content and the programmer for the technical components. This session will present the model, used by the authors, for developing and publishing research-based software.

Projects
Teacher Education and Needs Research

Needs Assessment: Level of Experiencewith Multimedia Computing Prerequisites

Loretta Cardinale
Norfolk, Virginia

Abstract

Abstract not available at press time.

The Computing Experiences of Teacher Education Students: A Longitudinal Study

Alison Elliot
University of Wollongong
NSW, Australia

Abstract

This project presentation reports on a three year study of the experiences of a group of preservice teacher education students as they undertook a major study in *Computers and Information Technology* as part of a Bachelor of Education program at an Australian University. Approached from a predominantly ethnographic perspective the study investigated student's entry level computing knowledge and experiences, learner responses and attitudes to subject matter and presentation, academic performance and achievement, and the impact of learning experiences on everyday encounters with computers, including those in classroom settings.

Discussion will focus on students' experiences and attitudes to learning *about* and*with* contemporary information technologies. Of particular interest are student's changing attitudes to the role of computers and information technologies in society and to learning with the support of computer-based tools over this three year period. At regular intervals over the duration of the course students wrote about their feelings and responses toward learning experiences in the form of a diary. The documentation of their learning experiences in this form provides unique and encouraging information on the gradual development of learner confidence and competence as the students, all women, progressed through the sequence of subjects.

Teaching Computer Educators: A Case for Logo

Sharon Yoder
University of Oregon

Abstract

Students in our master's program at the University of Oregon, are required to begin their program in the fall with a course in Logo. Increasingly it has become clear to us that this Logo course is a kind of "foundations" course for our computer education program. More than half of our students come from the Pacific Rim outside of the United States. Many of them come to us with limited experience with technology. They often see the computer only as a tool to deliver drill and practice or simple tutorials. As the term progresses, we see an ongoing change in all of our students. First they see the potential in Logo both as a programming language and as a learning environment. Through Logo, they begin to take a second look at education in general. They begin to question all aspects of education both in their own classrooms and in their home countries. Once these students understand the ideas behind Logo, it is easy for us to help them see the possibilities that technology offers. In the process of learning about Logo and working with Logo, students often experience for the first time the power of technology as a learning tool.

Projects
Multicultural Multimedia

Development of a Socio-Cultural HyperCard Stack for Criticism of Software

Barbara A. Boyer, Ph.D.
California State University
Los Angeles, CA

Abstract

A *HyperCard* stack focusing on socio-cultural factors has been designed to critique visuals in educational software. Socio-cultural factors can be defined as shared values, attitudes, and beliefs of specific groups of people. This sharing of belief systems and attitudes provides a basis for the socialization process and the way cultural assumptions about art and/or graphic designs become internalized (Boyer, 1987). Much that we learn in our own culture, particularly in visual communication, is at a taken-for-granted level and we are often not conscious that learning is even taking place.

Students and teachers need to be able to decode and analyze cultural visuals and interpret their meaning in both the larger context of society and within their own personal learning experiences.

In developing the *Socio-Cultural HyperCard Stack*, the author has examined visual images found in popular educational software products. A visual criticism strategy has been developed for identifying how role models of men and women are depicted, how different ethnic groups are depicted and how such topics as work and play are related to these depictions. Specific interactive media will be discussed with selected classroom activities provided for involving students and teachers in critiquing software.

References

Boyer, B. A. (1987). Cultural literacy in art: Developing conscious aesthetic choices in art education. In D. Blandy & K. Congdon (Eds.), *Art In A Democracy* . New York: Teachers College Press, 91-107.

Diverse-ALOT: A Multi-cultural, Multi-media Database

Darlene Nowak
Montclair Public Schools , California

Abstract

Montclair Public Schools is one of four K-12 recipients of Apple Corporation's *Apple Libraries of Tomorrow* grant. The Diverse-ALOT project was designed to promote interdisciplinary research in two magnet middle schools with a diverse student population. Teachers and students in Montclair needed resources to provide minority students with local heros and heroines as sources of inspiration and personal pride. Non-minorities need those same resources to cultivate appreciation for non-white, non-Western contributions to society.

This presentation will provide the audience with a general description of the project detailing the peaks and pitfalls of developing interdisciplinary, multi-media projects. A demonstration using samples of the product to date (locally produced laser disk, Hypercard databases, and Hypercard driven CD resources) will give educators a visual frame of reference when contemplating similar ventures. Handouts will outline specific information on contacts, resources, timelines and recommendations for districts wishing to replicate this project in whole or part.

The Conception and Design of Afrocentric Multimedia

Bruce Lincoln
Bank Street College
New York, New York

Abstract

The Black Inventors Continuum is an interactive multimedia software prototype. The Black Inventors Continuum represents a design exemplar of Afrocentric multimedia. More than simply utilizing Black imagery to exploit the visual features of Hypercard, the Black Inventors Continuum is based upon audiovisual communications systems derived fro an African cultural matrix. For the purpose of the project presentation, I will demonstrate the Black Inventors Continuum and dicuss the prototype as it relates to a new area of educational technology research and development that is underway at Bank Street's Center for Children and Technology—Afrocentric multimedia Within this context, the Black Inventors Continuum will be examinedaccording to the design principles that inform its structure, use and design.

Projects
Telecommunications and Science

Hypermedia Interactive Kaleidoscope for Educators in Science

Manorama Talaiver
Mathematics and Science Center
Richmond, Virginia

Abstract

Hike means to take a long walk or march a great distance for pleasure, for military training, or the like (Webster's Dictionary). HIKE at the Mathematics and Science Center (July 15-19, 1991) will be a walk in which science educators will enjoy and explore with technological tools for the purpose of enriching their instruction. Each letter in this acronym represents an enriching instructional strategy or a technological tool for making science learning more interesting, enjoyable, and real.

Telecomputing Support for Distance Learning

Inabeth Miller
Executive Director, MCET
Massachusetts Institute of Technology

Abstract

Abstract not available at press time.

Utilization of Computer Software for Athletic Training Competency Testing

Joanne M. Dunnock
Arizona State University

Abstract

The problems investigated in this research study were threefold. First, a computer management program was devised to evaluate athletic training competencies and provide feedback to the students. Second, a comparison was studied between traditional written testing and CMI testing. And finally, attitudes regarding CMI were evaluated.

The sample for the study consisted of 20 volunteer athletic training students. All of the subjects had completed a basic and an advanced course in athletic training. Each subject took a written test, participated in the CMI program, and completed the attitude questionnaire. The variables measured included previous computer experience, the written pretest scores, scores from the CMI program, and answers to a questionnaire regarding attitudes and the study.

Data were analyzed using a computerized statistical program and included the standard descriptive statistics, a t-test sampling, and an F-ratio.

There was a significant difference between the students performances on the written test and the CMI program. From the questionnaire, the subjects reported more enjoyment and better learning capabilities with the CMI project.

Special Presentation—SIGCUE
Legalities Notwithstanding, What's Fair in Software Use

A panel presentation with software publishers to clarify possible misunderstandings of the various schemes for software use in multiple machines.

Panelists

Chairperson:
Doris Carey
University of Colorado at Colorado Springs

Vendor representatives from:
Sunburst
Scholastic
MECC
Mindscape
Claris

Projects
Language Applications Including Multimedia/Hypermedia

The Proof is in the Pudding

Shelley Wepner
William Patterson College
Paterson, New Jersey

Abstract

Anecdotal and achievement data will be shared for a technology-based research study in reading. This year long study, funded in part by the New Jersey Department of Higher Education, was conducted with 85 eighth grade inner-city children from Paterson, New Jersey. These students were identified at the beginning of seventh grade as "dreamers", students who are sponsored by philanthropic New Jerseyans and whose dream of attending college can become a reality by additional support services.

Results of pre-posttest scores for a standardized reading achievement test, a New Jersey mandated writing sample, and attitudinal survey will be reported. Analysis of anecdotal data from student's written and oral comments also will be shared in terms of student's cognitive, affective, procedural, and psychological needs. Based on these findings, guidelines for future software development and use for this special population will be offered.

Developmentally Sequenced "Thinking Webs" for K-8 Hypermedia/Multimedia Materials Design

Judi Harris
University of Nebraska at Omaha

Abstract

Informational "webs" presented for exploration in hypermedia format are often designed without careful consideration of their developmental appropriateness. Interactive hypermedia packages may therefore be appealing to the *adults* recommending them for purchase, but may not be organized in ways that are as instructionally effective for *children* as these powerful technologies might allow. If there *is* an identifiable developmental sequence to the types of cognitive connections among content and process sub-categories that children make, it should be used to organize the design of curricularly-infused hypermedia/multimedia environments.

The first of a series of collaborative, mixed-method hypermedia design studies explored age-related patterns of children's associational thinking processes. The purpose of the study was to determine the nature and organization of content and process connections that students make among required curricular components. Data from large-group and individual student-created "concept maps" were analyzed and summarized according to types of cognitive connections reflected in approximately 400 students' written "thinking webs." The resulting developmental sequence of children's associational induction and its application to the design of developmentally appropriate interactive hypermedia materials will comprise the major portion of this project presentation.

Using *LogoWriter* as a Tool for Teaching Problem Solving

David Kressen
Polytechnic School
Pasadena, California

Abstract

Abstract not available at press time.

Projects
Social Studies Applications in the K-12 Environments

Explore, Discover, Present! Technology in the Social Studies Classroom

Peggy Healy Stearns
Software Evaluation Project
State University of New York at Buffalo

Abstract

Technology can turn students of social studies into enthusiastic explorers who navigate their own paths through a world of data. Students are not simply told history; they discover it. This is an important distinction because students have ownership and tend to remember what they discover.

For both students and teachers, the information explosion can be an exciting challenge that empowers us, rather than an onslaught that bewilders and intimidates us. Powerful tools for navigating this data are emerging, tools that encourage students to explore the world of information, to make sense of it, to form hypotheses and opinions, and to report their findings. Students become active participants shaping the direction of their own learning; teachers become facilitators guiding enthusiastic learners on an exciting shared journey.

This presentation will provide an overview and demonstration of exciting new software that encourages students to explore social studies, to create dynamic displays of statistical trends and relationships, to engage and challenge their higher order thinking skills and to organize and present their findings. Specific examples will include the use of simulations, visual databases, videodisc, hypermedia environments, and programs for the publication and presentation of data.

Integration of National Geographic Kids Network Across the Curriculum

Susan Short
Hebrew Academy
Westminster, CA

Abstract

This presentation will introduce various practical methods of integrating National Geographic Kids Network "Hello" Unit across the curriculum in a middle school setting. It will show the use of cooperative learning groups and how computer technology enhanced the "Hello" unit for my students. Science classes gather and interpret data. During social studies, students studied global addresses, practiced map skills, and compared their own community with others on their "Hello" team. In language arts the students wrote individual letters about themselves, their school and their community, a class letter, and creative pieces about pets. In math class the "Hello" data was used to create student graphs and word problems. The unit was culminated with an art project. Students used National Geographic software, Printshop, Crossword Magic, Children's Desktop Writing and Publishing Center and FrEdwriter to complete the "Hello" unit. Examples of these projects will be shown.

Contemporary Issues in Social Studies Utilizing the Latest Technology

Deidre Foreman
Calcasieu Parish School Board, Louisiana

Abstract

This year the latest current events information is beamed via satellite technology into Calcasieu Parish high school for a new one-half unit of credit social studies elective entitled "Contemporary Issues". Utilizing the latest technology, a multimedia arrangement, teachers and students explore, interpret, and evaluate current issues on the local, state, national, and international levels. They also use newspapers, magazines, television programs such as CNN Newsroom and C-Span and X-PRESS Information Services. Local cable companies have provided a cable outlet in each room in addition to the X-PRESS Information Services.

Special Presentation
Multimedia in the Classroom Today

Roger Wagner
Roger Wagner Publishing
El Cajon, California

Abstract not available at press time.

Special Presentation—SCS
The Algebra Workbench: A Tool for Inquiry

The Algebra Workbench: a Tool for Inquiry Mathematics in Pre-Algebra and Algebra

Solving algebra problems can be thought of as having two aspects; first are strategic issues of choosing what operations to do, and second are symbol manipulation issues of how to carry out a particular operation. Many algebra students find trying to do both of these things at once to be overwhelming. The algebra workbench is a tool that tries to split these two aspects of solving equations allowing the student to focus on the strategic aspects of choosing operations and allowing the computer to handle the symbol manipulation. The algebra workbench is an intelligent algebra calculator that allows students to try out a strategy, see the results, and then decide to follow it or try a different path. The workbench also includes a variable expert that can be set to give advice at various levels of sophistication. The algebra workbench can handle a variety of equations from linear to trigometric to simultaneous equations as well as inequalities. It also includes a graphing capability. We will demonstrate it and discuss its use in an inquiry oriented approach to algebra.

Panelists:

Ricky Carter
Bolt, Beranek, and Newman
Cambridge, Massachusetts

John Richards
Bolt, Beranek, and Newman
Cambridge, Massachusetts

Bruce Mallory
Lexington Public Schools
Lexington, Massachusetts

Projects
Learning Styles and Multimedia

The Hypertext Vygotsky Project
H. Eric Branscomb
Director, Writing Program
Stonehill College
North Easton, Massachusetts

Abstract

One of the most influential cognitive psychologists of the twentieth century is Lev Vygotsky, a Soviet researcher known for his studies of concept formation and language development. His research in the 1920's with young children led to the development of what have come to be known as "Vygotsky Blocks", a set of multi-colored and multi-shaped blocks used to understand how children group objects, thus forming concepts of various degrees of maturity, and the role language plays in such groupings.

With the advent of hypermedia, it is now possible to illustrate Vygotsky's theories on the computer. This project is a hypertext-based manual of Vygotsky's principles of concept formation, cognitive development, and the mediation of language in cognitive development. The hypertext illustrates the use of Vygotsky Blicks in action, and it illustrates the role of language in concept formation and super ordination for older children and adults.

Teaching HyperCard to Children

Maria Swayne and Karen Hare
Los Angeles Unified School District, California

Abstract

HyperCard™ is a very versatile classroom tool. It has proven successful for creating teacher-generated lessons, for classroom management, and for laser picture management. This workshop will share lessons that teach HyperCard basics to elementary and junior high students, enabling them to create fairly sophisticated stacks of their own. HyperCard programming is an excellent tool for developing students' critical thinking abilities. It teaches flexibility in problem solving. Students working in groups learn brainstorming techniques and cooperative learning methods.

In addition to sample lessons, the presenters will also demonstrate several successful HyperCard stacks that they have created, including one that teaches matrix logic, and an interactive video whole-language biology lesson.

The presenters are a part of the California Model Technology Schools Project/Bell Complex in Los Angeles County. They have presented at CUE Santa Clara earlier this fall. They are excited about the endless educational possibilities of HyperCard, and are anxious to share some of their most successful ideas.

Image Processing for Teaching

Richard Greenberg, Robert Strom, Steven Croft, and Lyn Doose
Lunar and Planetary Laboratory, University of Arizona

Abstract

Many students, perhaps most, are visual learners. For such people, manipulation of images may provide an attractive entree into science and mathematics, especially for those whose linguistic or cultural background has been an impediment to traditional coded (language-based) teaching. Certainly, in the formal sense, images convey information to the brain orders-of-magnitude faster than words or other codes. For all students, learning through image manipulation can complement and enhance language-based instruction.

Image processing allows pictures of photographic quality to be manipulated through a computer. The Image Processing for Teaching project at the University of Arizona is educating teachers in the technique and technology, and in the scientific content of a wide variety of data sets. The teachers are developing activities for children in a wide range of grade and subject areas. The project is supported by the National Science Foundation, Apple Computer's ACOT Project, the University of Arizona, and participating school districts.

Paper
Desktop Publishing: A Course Reader (Desktop publishing or desktop plagiarizing?)

Paul Matray
Stevenson School
Pebble Beach, CA

Abstract

Desktop Publishing (DTP) software has given the teacher the capability of producing course readers which look as if they were professionally published. It literally puts at the teacher's fingertips the power to publish first-rate classroom materials. With this power there are some needed cautions to insure that the technology does not infringe on the rights of people who originally had their work published by publishing houses and are now being plagiarized by over aggressive neophyte/amateur "publishers." The procedures by which an aspiring amateur writer/publisher can produce a professional looking course reader are described. The Apple Macintosh computer, several software programs and the use of photocopy shops are discussed in terms of how they can be brought together to produce a course reader at a reasonable price by any teacher who has the desire to be *published*.

Introduction

Teachers communicate a significant amount of knowledge through the written word. Unfortunately they are often limited by the books provided by publishing houses. With the advent of the personal computer, desktop publishing (DTP), and photocopying stores with binding capabilities, teachers can now produce their own pamphlets, books, manuals, study guides, and/or any other lengthy document (all of which will be collectively referred to as *course readers* in the remainder of this paper). The technology has provided possibilities which did not exist just three years ago. Before that time, teachers were forced to utilize the ditto machine, a hole punch, and a ring binder as the only method of producing a compilation of course materials in a single document. The end result may have had very good content, but its presentation detracted from the endeavor. The students often viewed them as a poor substitutes for bound volumes.

Copy machines have made it easier to produce better-looking materials with a classier appearance. They also provide the opportunity to photocopy entire pages of material verbatim and include them as part of a longer document or to simply distribute them to students as individual entities. Most teachers realize that there are copyright concerns whenever one photocopies a copyrighted document. Each of us deals with this situation in our own way. Often we cite the originator of the material and we do not sell the material at a profit. We are, nonetheless, impacting the ultimate sales of that material by the publisher.

With the advent of DTP, teachers can now take the final step and collect a series of materials which meet their own personal needs, organize them in a final document and finally have them bound at their local copy shop. The finished product can, for all intents and purposes, look like a paperback book produced by a publishing house. I would hope that my colleagues in the teaching profession will use this new found power to publish in an appropriate way and not misuse or abuse their newly acquired power. Teachers should produce their own materials and not simply copy other people's work. There is a tremendous tendency to simply copy the information one wants to use and put it in the desired order and call it ones own, either implicitly or explicitly.

In this paper I will describe a process by which teachers can produce course materials such as lab manuals, booklets, study guides—*course readers*. Most of us have thought of how desirable it would be to have course readers which included only the information and materials we would be presenting to our classes. Because such books did not exist, we had to buy several books/manuals and, in addition, copy considerable supplementary material. As teachers, we are compelled to use lab manuals or books which contain some of the assignments we want to use in our classes, but much of what is included does not meet our needs. With the advent of the personal computer and DTP software, this is no longer the case, we now have the ability to be our own publishers!!

There are a number of specific reasons why course readers are desirable documents to create for your classes:

- you have total control of the content;
- it saves wear and tear on the school's copy machines;
- it saves you time in the future;
- it organizes your courses for the students and possibly yourself;
- you avoid the copier room; someone else does your copying as they publish your course reader;
- all handouts are bound in one volume and thus more difficult for the students to lose and easier to find; and
- yearly revisions are much easier.

As the years go by, it may become increasingly common for teachers to even write their own books. This will become most obvious at the collegiate level where professors have the ability to write specialized books for a limited audience and have them published by their own means with a professional look at a reasonable cost. Even now, citations at the end of scholarly journals begin to include the term, "Privately Published," when citing sources.

Getting Started

During the summer of 1989, I "published" course readers for my Biology and Advanced Placement Biology classes at the Stevenson School in Pebble Beach, California. This past summer I not only updated the biology manuals, but also aided the U.S. History teachers at Stevenson in "publishing" a course reader for their classes.

Stevenson School is an independent resident and day school. It is located on two campuses—K-8 on the Carmel campus and 9-12 on the Pebble Beach campus. I am a biology teacher at the high school where I have taught biology for the past 16 years. I have been using the Macintosh computer for the past three years and have learned to use *PageMaker* over the past two years. During this time I have produced a number of publications for the school including the Parent Guide, Student Handbook, Faculty Guide, football programs, music programs, commencement programs, calendars, etc. (Matray, 1990).

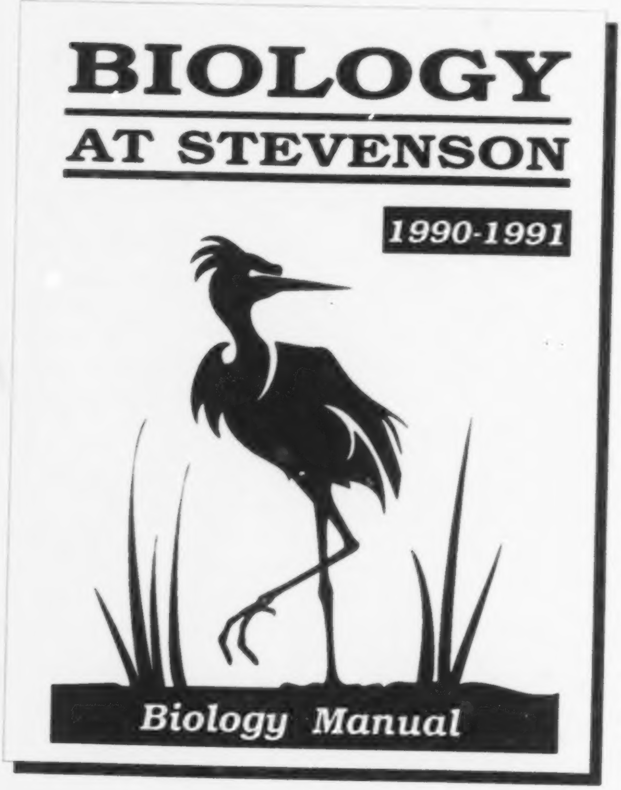

BIOLOGY
AT STEVENSON
1990-1991
Biology Manual

Figure 1. Cover of Biology Manual

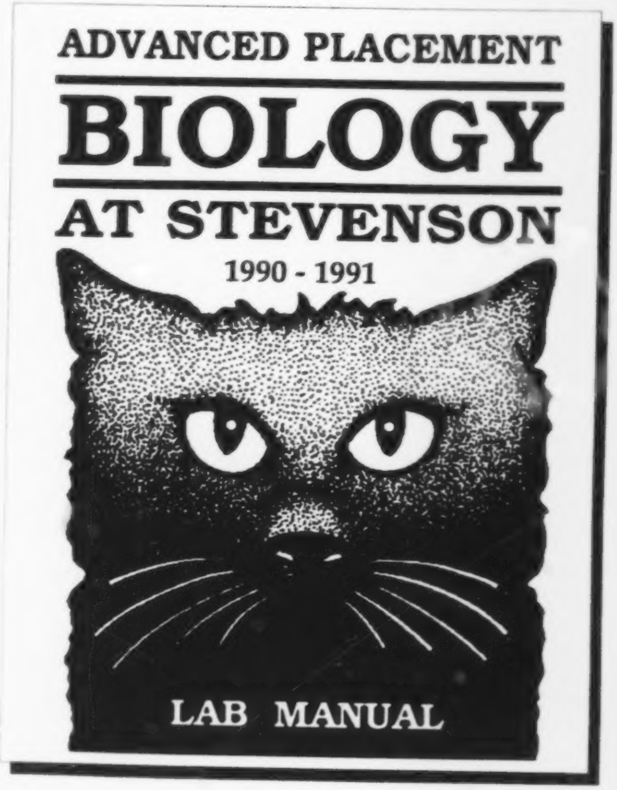
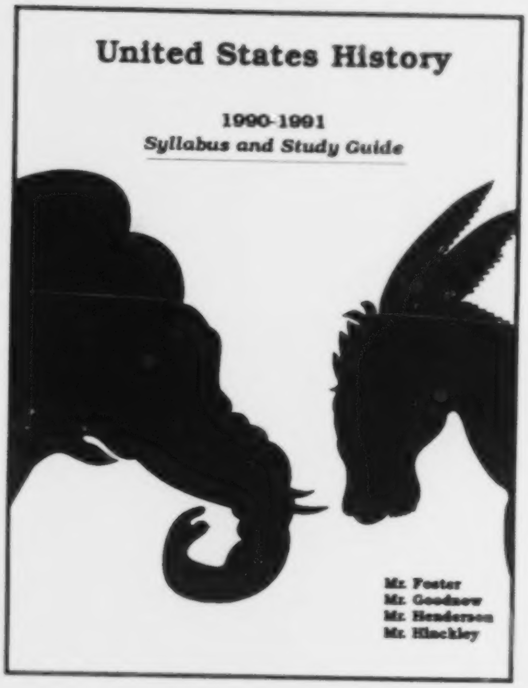

ADVANCED PLACEMENT
BIOLOGY
AT STEVENSON
1990 - 1991
LAB MANUAL

Figure 2. Cover of Advanced Placement Biology Manual

Over time the Macintosh computer has made it possible for teachers to become increasingly efficient in their gradebook responsibilities and paperwork. Teachers can now keep tests, homework assignments, parent letters, etc., on disk and then simply revise them on an annual basis. They are now ready for the next step—to become their own publishers.

Teachers spend a considerable amount of time producing handouts for their classes which are commonly distributed on an irregular basis and are often misplaced or lost by students. Some teachers have actually provided binders which include a number of copied pages and can be updated periodically. These efforts can range from a rather organized system to a rather "catch-as-catch-can" situation.

I have found that my teaching life has become less hectic and more organized in the past two years since I began providing my students with manuals at the beginning of each year. I am no longer running to the copier on a daily basis to run off handouts. I still have to reproduce tests and finals and other items which come up during the year, but I now have the syllabus for each of the three terms included, review sheets, genetics problem sets, lab assignments, diagrams, questions,



United States History

1990-1991
Syllabus and Study Guide

Mr. Foster
Mr. Goodnow
Mr. Henderson
Mr. Hinckley

Figure 3. Cover of U.S. History Study Guide

etc. This not only gets me more organized, but in turn forces students to get more organized. If students lose their manuals, they simply go to the bookstore and buy another copy. I am no longer responsible for providing replacements.

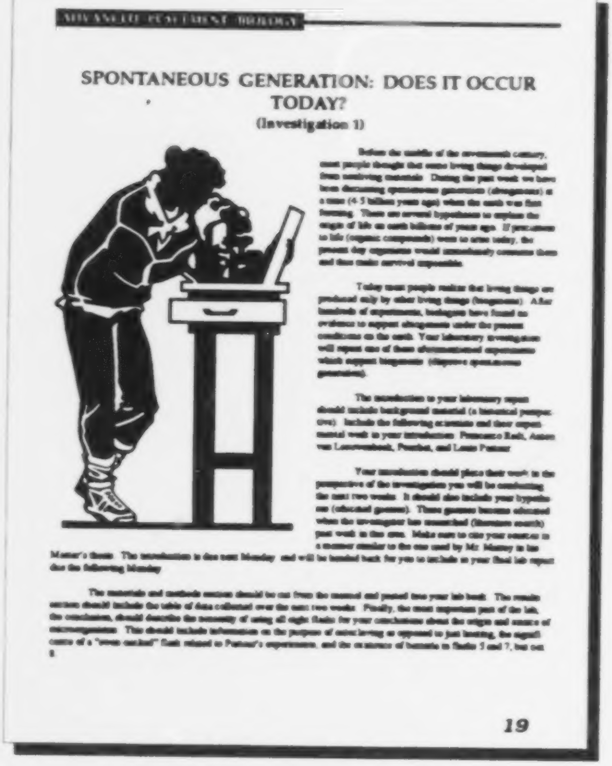
Hardware

It is my opinion that the Macintosh computer is the computer of choice at the secondary or college level. It is easy to learn and has excellent graphics capability. The three covers of the manuals (see figures 1, 2, and 3) which I prepared this past summer are representative of the font and graphic capabilities possible with the Macintosh.

Many secondary and post-secondary schools now have Macintosh labs on campus. At Stevenson School we have a word-processing room where typing skills are taught to any student entering the upper school who cannot type at 40 words per minute. This lab has 18 Macintosh Plus computers networked to four Imagewriter II printers. We also have a publications room where we have a Macintosh IIcx, an SE30, and an SE20. This room also contains a Laserwriter IINT and a Hewlett Packard Scanner. In that many schools have invested in Macintosh computers in the past 5-6 years, the concerns regarding hardware costs may be minor.

If your school has not yet recognized the power of the Macintosh, it may be necessary for you to convince them to purchase a basic set-up so that you can have the necessary equipment to produce your manuals. I would recommend the purchase of a Macintosh computer with a hard drive. A Macintosh Classic computer can be purchased at the present time for $799 (educator buy price) and a 20 mb hard drive from a third party for $299. A Macintosh Classic which has an internal 40 mb hard drive costs $1199. Either computer would be a reasonable choice and would allow you to do your word processing and page layout. The advantage you derive from a more expensive system is that you will have greater speed, and if you choose, a large screen monitor. Your decision will probably be based upon whether or not your business manager is a "generous and understanding" individual and/or if you are personally paying for the system.

Besides the actual computer, the purchase of a laser printer and scanner would allow you to produce camera ready materials which contains scanned images. The Apple Laserwriter IINT costs $3365 and will print Postscript—you want Postscript, *trust me*. There are several other laser printers available on the market, which have received favorable reviews (Fraser, 1990; Heid,1990). Reading *recent* hardware and software reviews in *Macworld* and *MacUser* is an excellent way to increase



ADVANCED PLACEMENT BIOLOGY

SPONTANEOUS GENERATION: DOES IT OCCUR TODAY?
(Investigation 1)

19

Figure 4. Page from Biology Manual

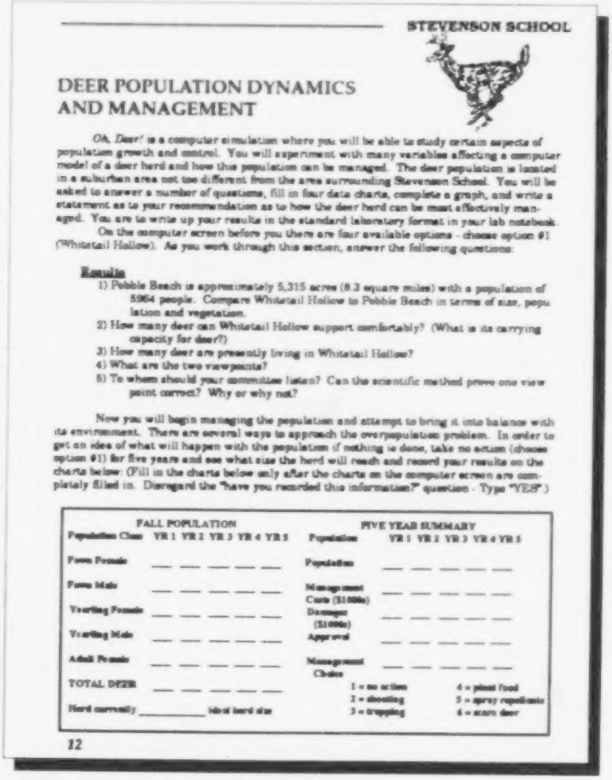
BEST COPY AVAILABLE

the likelihood that your money or your school's money is wisely spent. If the price of a laser printer is out of reach, fear not. You can go to a local service bureau and make use of their laser printers. Most communities have several of these establishments. In the Monterey area Kinko's is the one that I have noted; I know that this company is a world-wide organization. They charge a fee for the use of their laser printers, but their fees do not currently approach a fraction of the purchase price of a new laser printer.

Scanners are also handy pieces of hardware to utilize when putting together a manual. You can scan in images (like your school's logo) which are line art or make use of photographs which you have located or taken yourself. We have a Hewlett Packard JetScan and have been most impressed with its capabilities. The September 1990 issue of *MacUser* reviews 29 scanners (Linzmayer, 1990). You can also probably find a service bureau where images can be scanned at a reasonable charge. Stevenson bought a Hewlett Packard scanner in June 1990 for about $1500.

Software

If you are not already using a word processing program, and you are attending this conference, you must have gotten on the wrong plane. It is of paramount importance to have your prose on disk. If you have been using a Macintosh for a number of years and have saved your labs, worksheets, problem sets, review sheets, etc. you have already completed a good portion of your first manual. Even if you have them saved in *AppleWorks*, the transfer to Microsoft *Works* is as simple as a few key strokes. If you have chosen an MS-DOS based system (why would anyone do that?), you can still transfer the data over, by using *MacLink* with a comparable number of keystrokes, if your data is on a 3.5" disk and the Macintosh has an FDHD floppy drive. Most word processing files can be loaded into a page layout program. An inexpensive word processing program which will run on a single drive Macintosh Plus is *WriteNow*—it is very easy to learn and use.



STEVENSON SCHOOL

DEER POPULATION DYNAMICS
AND MANAGEMENT

Oh, Deer! is a computer simulation where you will be able to study certain aspects of population growth and control. You will experiment with many variables affecting a computer model of a deer herd and how this population can be managed. The deer population is located in a suburban area not too different from the area surrounding Stevenson School. You will be asked to answer a number of questions, fill in four data charts, complete a graph, and write a statement as to your recommendation as to how the deer herd can be most effectively managed. You are to write up your results in the standard laboratory format in your lab notebook.

On the computer screen before you there are four available options - choose option #1 (Whitetail Hollow). As you work through this section, answer the following questions:

Results

1) Pebble Beach is approximately 5,315 acres (8.3 square miles) with a population of 5964 people. Compare Whitetail Hollow to Pebble Beach in terms of size, population and vegetation.
2) How many deer can Whitetail Hollow support comfortably? (What is its carrying capacity for deer?)
3) How many deer are presently living in Whitetail Hollow?
4) What are the two viewpoints?
5) To whom should your committee listen? Can the scientific method prove one viewpoint correct? Why or why not?

Now you will begin managing the population and attempt to bring it into balance with its environment. There are several ways to approach the overpopulation problem. In order to get an idea of what will happen with the population if nothing is done, take no action (choose option #1) for five years and see what size the herd will reach and record your results on the charts below: (Fill in the charts below only after the charts on the computer screen are completely filled in. Disregard the "have you recorded this information?" question - Type "YES".)

FALL POPULATION

Population Class	YR 1	YR 2	YR 3	YR 4	YR 5
Fawn Female	___	___	___	___	___
Fawn Male	___	___	___	___	___
Yearling Female	___	___	___	___	___
Yearling Male	___	___	___	___	___
Adult Female	___	___	___	___	___
TOTAL DEER	___	___	___	___	___

Herd currently ________ Ideal herd size

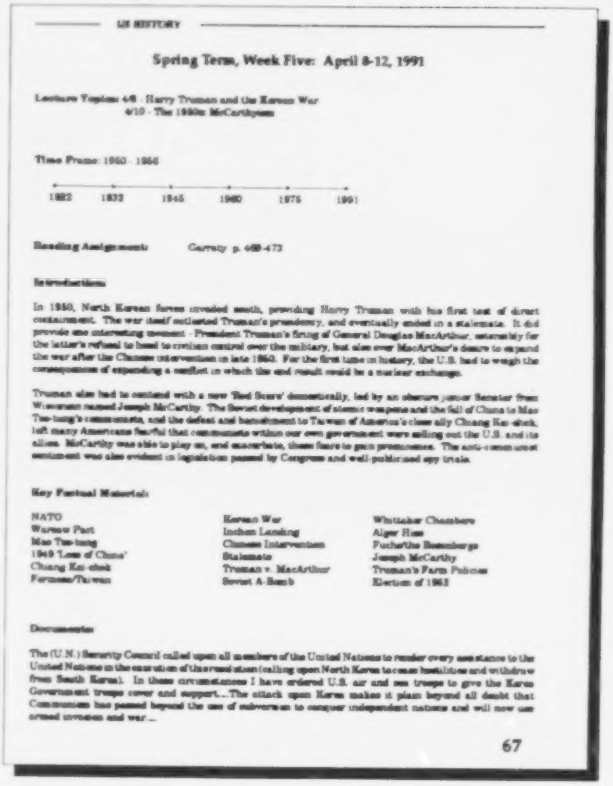
FIVE YEAR SUMMARY

Population	YR 1	YR 2	YR 3	YR 4	YR 5
Population	___	___	___	___	___
Management Costs ($100s)	___	___	___	___	___
Damages ($100s)	___	___	___	___	___
Approval	___	___	___	___	___
Management Choice	___	___	___	___	___

1 = no action
2 = shooting
3 = trapping
4 = plant food
5 = spray repellants
6 = scare deer

12

Figure 5. Page from A. P. Biology Manual



US HISTORY

Spring Term, Week Five: April 8-12, 1991

Lecture Topics: 4/8 - Harry Truman and the Korean War
4/10 - The 1950s McCarthyism

Time Frame: 1950 - 1954

1882
1932
1945
1960
1975
1991

Reading Assignment: Garraty p. 460-473

Introduction:

In 1950, North Korean forces invaded south, providing Harry Truman with his first test of direct containment. The war itself outlasted Truman's presidency, and eventually ended in a stalemate. It did provide one interesting moment - President Truman's firing of General Douglas MacArthur, ostensibly for the latter's refusal to bend to civilian control over the military, but also over MacArthur's desire to expand the war after the Chinese intervention in late 1950. For the first time in history, the U.S. had to weigh the consequences of expanding a conflict in which the end result could be a nuclear exchange.

Truman also had to contend with a new 'Red Scare' domestically, led by an obscure junior Senator from Wisconsin named Joseph McCarthy. The Soviet development of atomic weapons and the fall of China to Mao Tse-tung's communists, and the defeat and banishment to Taiwan of America's close ally Chiang Kai-shek, left many Americans fearful that communists within our own government were selling out the U.S. and its allies. McCarthy was able to play on, and exacerbate, these fears to gain prominence. The anti-communist sentiment was also evident in legislation passed by Congress and well-publicized spy trials.

Key Factual Material:

NATO
Warsaw Pact
Mao Tse-tung
1949 'Loss of China'
Chiang Kai-shek
Formosa/Taiwan

Korean War
Inchon Landing
Chinese Intervention
Stalemate
Truman v. MacArthur
Soviet A-Bomb

Whittaker Chambers
Alger Hiss
Fuchs/the Rosenbergs
Joseph McCarthy
Truman's Farm Policies
Election of 1952

Documents:

The (U.N.) Security Council called upon all members of the United Nations to render every assistance to the United Nations in the execution of this resolution (calling upon North Korea to cease hostilities and withdraw from South Korea). In these circumstances I have ordered U.S. air and sea troops to give the Korean Government troops cover and support...The attack upon Korea makes it plain beyond all doubt that Communism has passed beyond the use of subversion to conquer independent nations and will now use armed invasion and war...

67

Figure 6. Page from U.S. History Study Guide

You definitely need a desktop publishing program (DTP). There are those who suggest that a sophisticated word processing program like Microsoft *Word* and/or *Macwrite II* can do a similar job; I would argue that these programs simply lack the power to produce the more complex page layout work which will give your final document that professional look. *PageMaker 3.02* is the DTP program I use and I am now just starting to make use of *PageMaker 4.0.*, *Quark Xpress* and *Ready, Set, Go!* are the other two DTP packages that people most frequently use for computer layout on the Macintosh. Taub (1989) com-

BEST COPY AVAILABLE

pares the three and presents the advantages of each. If your school newspaper is done on a Macintosh, they are probably using one of these programs and you would therefore already have ready access to both the software and skilled personnel who could help you learn the program. If your school does not own a DTP program, you can get an educator buy price from Aldus for much less than the retail price. We bought 5 copies of *PageMaker 4.0* last summer for $450 whereas the retail price for one copy is now $795!!!

Graphics are important in your final document only if you want your students to read it. It is my opinion that you improve the appearance and readability of the final document by breaking up the text with images. It should look more like *USA Today* (done on the Macintosh) and less like the *Wall Street Journal*. It's a fact of life that our students were brought up in the television age and they are visual learners. If you prefer maintaining your "high standards" and making your manual look like a scholarly piece of work, DTP can satisfy your desires. Simply, I feel that there is a greater likelihood that more students will be drawn into a document which has been enlivened with graphics, than one that is simply text.

Stevenson owns several graphic packages which it uses in various DTP projects. Aker (1988), Fenton (1989,1990), and McCloy (1989) each present a comprehensive guide to what is available and their opinions of which packages are good buys. In the figures which accompany this paper, you can see how I have made use of these graphic images. They can be overused; you need to use appropriate graphics in appropriate contexts. I like Metro Imagebase, 3G Graphics, T-Maker, Pro Art, Dynamic Graphics, Adobe Collection, and Dubl-Clik Software. In addition, if you have access to a scanner, you can produce your own graphics by simply scanning them in and then modifying them in any way you wish.

Layout

> "Thanks to desktop publishing software, millions of people now have the tools to create gaudy, unbalanced publications full of klutzy layouts and mismatched fonts. Just as easy-to-use financial software won't instantly make you a CPA, accessible desktop publishing tools won't magically give you taste." (Naiman, 1988)

In order to create a course reader which looks published, it is necessary to practice good common sense when doing your layout. It also helps to have an eye for balance and practice a few basic rules. Bookstores carry a number of books which are written for people who are using DTP software. The books describe the principles used by layout professionals who have used these techniques, even before the computer was used in their profession. It would be a good idea to visit the computer section in your local bookstore to peruse some of these books before you begin your layout work. *Publish!* magazine is a monthly available on the magazine racks and devotes a good portion of each issue to the subtleties of layout.

I have included in this article several pages from the three course readers we now publish at Stevenson (Figures 4, 5, and 6.). They give some idea of how a finished document might look. I created a consistent look on each page by putting a header with the name of the course and the school. The footer included the pagination which is done automatically in page layout programs. I also tried to maintain a consistency in the titling of labs or other assignments which occurred on a regular basis in the course reader. Font typefaces, styles, and sizes should be kept to a minimum in order to avoid what is described in the above quotation. The U.S. History reader did not have any graphics and only took about five hours to layout, whereas the biology reader included a number of graphics and as a result took much more time to complete.

Publishing

Once the document is camera-ready, it can be taken to an offset printer or to a copy shop. One of the key factors influencing the decision is cost. Offset printers can print large runs at a much lower per copy price than can copy shops. Since we produce anywhere from 50 to 150 copies, I have used Kinko's copier shop, even though I have priced the work at several offset print shops. I found that it is cost effective to utilize an offset printer if more than 400-500 copies are produced.

The key considerations are type of paper, cover and binding. I have found that a 60 lb. paper is the best for double sided documents and have used plain white. In order to create a more formal look, a textured and/or colored paper can be used. It is my opinion that the binding procedure is the most critical aspect of the work done by the print shop. You have control over choice of paper, and the print shop usually does a good job copying. There are basically four methods of binding which are commonly available at print shops: staples, Velobind, Cheshire bind, and plastic rings. We used Velobind the first year thought it was the most substantial, but the students proved us wrong. Staples can be sturdy but they are unattractive when used to staple 40 sheets of 8 1/2 x 11 paper together. Plastic ring binding is not substantial enough and it can be easily pulled apart. This past year we decided to use the Cheshire bind which is a glued binding allowing the manual to lie flat when opened. We have been pleased with its durability and I like its appearance. It looks like a professionally produced booklet and gives our work the look of a professionally published document. The cover was made of index stock paper and a different color was used on each of the three manuals.

It is important to shop around and get bids from different copy shops if you are interested in getting the best price. Some school districts even have their own print shops which may be able to undercut commercial shops. The cost of the final document is based mainly upon the length of the document and quantity printed. Kinko's agreed to copy each page for 4 cents including paper and then charged $1 to bind each unit. This meant that our 80-90 page documents cost each student between $4.20 and $4.60 each. Once I delivered the originals to the print shop and made the aforementioned decisions, I was done with my part in the endeavor. The print shop delivered the 350 booklets to the school 24 hours later!! This is another advantage over using an offset printer which usually requires a 1-2 week turnaround time. Because we are an independent school where our students buy their books, it was easy to simply sell, at cost, the finished manuals directly out of our bookstore. If a student loses his manual he needs to purchase another one. I have relieved of the responsibility of venturing to the copy room to replace lost copies.

Conclusion

Teachers now have the ability to compile a professional looking, camera ready document in their own office and then have it published at a local copy store at a *reasonable* cost. If the teacher types in his/her material, he/she can then either use a page layout program himself/herself to produce the final camera ready material or simply hire someone who knows DTP to complete this step in the process. I have been in both roles — completing the entire document myself and assisting a colleague in the layout phase. If teachers ever hope to be viewed as professionals, it will be necessary for them to produce professional level work. The computer should be viewed as a powerful ally in their on-going search for professional status.

References

Aker, Sharon (1988) You gotta have art. *MacUser 4*(8),180-190.

Fenton, Erfert (1989) The clip art buyer's guide. *MACWORLD 6* (10), 138-145.

Fenton, Erfert(1990) Mac art mart. *MACWORLD* 7(10), 194-201.

Fraser, Bruce (1990) Buyer's guide: the letter-perfect postscript printer. *MacUser 6*(10): 89.

Heid, Jim (1990) Page printers revisited. *MACWORLD* 7(10), 184-193.

Linzmayer, Owen (1990) An inside look at scanners. *MacUser 6*(9), 132-190.

Matray, Paul (1990) *Desktop Publishing in a Secondary School*. Computers in Education. Proceedings of the IFIP TC 3 Fifth World Conference in Education—WCCE 90. Sydney, Australia, July 9-13,1990. Elsevier Science Publishers. Amsterdam, the Netherlands.

McCloy, Marjorie (1989) Is it art yet? *Publish! 4*(7),46-52.

Naiman, Arthur (1988) *The Macintosh Bible*. Publishers Group West, Emeryville CA.

Taub, Eric (1989) Room to grow. *Personal Publishing 5*(7), 51-59.

Paper

Computers, Writing, and Whole Language: Staff Development Issues

Cathy L. Gunn
Northern Arizona University

Abstract

This paper describes an ethnographically informed case study of one first-grade classroom. The study sought to describe the developing writing environment when computers were introduced to a novice computer-using teacher and her students. A computer consultant, also the researcher, introduced and helped implement a computer as a writing tool within a Writer's Workshop format. Journal writing, field notes, written artifacts, written questionnaires, and interviews were utilized to gather data.

Data demonstrated that first-grade children can indeed write using a word processor. Word processing functions such as DELETE and SAVE were used by first-grade writers to edit and to add on to their stories at a later time. It was also found that children's perceptions of what computers are used for may influence their own use of the computers.

Observations of the teacher's process and struggle in learning to use the computer herself for writing and at the same time integrate the computer into their Writer's Workshop suggest that computer staff development is critical when a novice computer-using teacher introduces computers to children for writing.

This research suggests implications for school computer policy, especially in the areas of staff development and the integration of computers into existing curricula. Although this study describes only one teacher's process and one classroom, it reveals the complexity of technological change and school culture. As this study progressed, patterns emerged that have implications for staff development training. Information from a review of literature combined with findings from this study suggest changes in existing computer inservice and a computer staff development model is offered.

> Sometime, during the second half of the 20th century, Western Society evolved from an industrial to a post-industrial, or to be precise, an Information Society. Education for an industrial society centered on teaching the Three R's: "Reading, 'Riting and 'Rithmetic." Its aim was to produce a disciplined workforce-punctual, conformist, specialized-to operate the brute machinery of the nation-state. Education for the information society will center on the Three Cs: "Children, Computers and Communication." Its aim will be to produce a creative workforce-adaptable, entrepreneurial, interdisciplinary-to help solve the problems of this planet. (Stonier & Conlin, 1985, prologue)

Many people believe that our society is in the midst of a technological revolution, the essence of which is change (Simonson & Thompson, 1990). Similarly, scholars hailed the invention of the printing press as a technological breakthrough that changed and reshaped society. Gutenberg's printing press may have seemed threatening to many people after its invention: perhaps there would be no need for the quill, or for hand copiers of books, or even teachers because students would have access to books. The computer has created similar concern today. There is a fear that the computer will replace many people in the work force. It is not the invention, however, that changes society and thus, education, but people who are able to grasp the possibilities inherent in the innovation and effectively act on their insights to institute change.

The computer is a potentially great invention for our society. Cultural conditions and expectations inform the acceptance or rejection of this invention and the extent to which it is used in the educational environment. In particular, this acceptance or rejection may well depend on the imagination of the society's educational leadership.

Imaginative educational leaders, who have a vision of an Information Society and perceive education centering on the "Three C's" (Children, Computers, and Communication), are providing computers for each classroom. Classroom teachers, however, may not have the choice to accept or reject this technological invention known as the computer. Although computers are provided, they are not necessarily being implemented effectively. Computers often sit unused or they are used for drill and practice software only. Many computer educators criticize its overuse and the fact that drill and practice is not the best use of the computer's power. While research exploring the effectiveness of the computer as an educational tool for younger children is far from definitive, word processing is beginning to show promise as a powerful tool for beginning writers (Hoot & Silvern, 1988).

With the introduction of computers and word processing into classrooms comes the need for workshops and inservice opportunities to prepare teachers to work with new technology. This is particularly true for teachers employing a whole language approach to literacy, with its accompanying Writer's Workshop to facilitate and develop the writing process. Whole language is a philosophy that refers to meaningful, real, and relevant teaching and learning (Routman, 1988). This philosophy follows the premise that children learn to write and read much the same as they learn to talk: by experimenting with language within a supportive environment where adults accept approximations and scaffold the learning process. Learning activities are open-ended and involve student choice, discussion, and sharing in a social, literate environment. A Writer's Workshop format provides a specific time within a school day in which children draft stories, conference with peers and their teacher, revise and edit their work, choose writing to be published, and share their published writing.

The computer can be used as a writing tool that promotes this whole language philosophy, but writing teachers need to learn how to use word processors before they are able to explore pedagogical approaches appropriate for their students (Herrmann, 1989; Simonson & Thompson, 1990). Unfortunately, learning to use computer applications is a time- and energy—consuming project.

Many classroom teachers lack knowledge of computer use and are often at a loss to know how to use and integrate the computer in their classroom. It is necessary to think about how teachers' theoretical assumptions and beliefs about language and literacy acquisition, and writing in particular, influence their view of the role the computer plays as a writing tool. In addition, other factors that need to be addressed include teachers' and young children's perceptions of what computers can do and also the computer's potential role in the writing process.

Word processing offers tremendous possibilities for education. Computer functions that allow the writer to move or delete text easily can help educators unleash some of the computer's power in the writing classroom. Word processing also has the potential to improve the teaching and learning of writing. Research concerning the effects of word processing on the writing process is still not definitive, but studies have begun to suggest that student attitudes toward writing may improve with word processing, students may be more willing to revise with computers than with paper and pencil, and students may spend more time writing when they use a computer (Daiute, 1986; Hawisher, 1989; Kahn, 1987).

Before teachers can use a word processor in a writing classroom, they must first become comfortable using the computer. Becoming comfortable with the computer may be only the first step in introducing the computer to children, however, because teachers also must make basic decisions about the role the computer will play in their classrooms. Such decisions will necessarily be made on their understanding of the use of a computer as a writing tool within the writing development of the young children they teach. It is important that their frame of reference includes what is already known about how children learn.

This ethnographically—informed case study of one first—grade "whole language" classroom sought to describe the developing writing environment when computers were introduced to a novice computer-using teacher and her students. As the computer consultant, and also the researcher, I introduced and helped implement the computer as a writing tool within a whole language classroom. Six months were spent in the classroom. Beginning in December, 1989, I introduced first-grade teacher, Lynn Tenille, to word processing in a one-on-one setting once a week after school, and together we developed a procedure to introduce each of her first-grade students to writing with a word processor. I was also an observer in the classroom three mornings a week to become familiar with Lynn, as the teacher, the children as writers, and the classroom as a writing community. In addition, I also participated as a teacher's aide during Writer's Workshop time. In early March, 1990, students were introduced to computers for writing, and the study continued until the end of school in mid-June. The study reported here covers the period spent working directly with the teacher and her students, both at their desks and at the computers.

All subjects, both teacher and students, had been exposed to the computer for drill and practice software, but on a limited basis. None of the students had used a computer for writing, and their teacher, Lynn, had no previous computer instruction. At the beginning of the study, she had never used the computer for writing herself and knew little more about the computer than how to turn it on.

The teacher's developmental process was documented through journal writing and audio-recordings. Student behavior at the computer was documented by researcher field notes and audio-recordings of conversations. Writing samples, both pencil and paper and computer printed, were collected and compared. Personal interviews of the students were employed to assess their attitudes about writing, uses of a computer in the adult world, and the use of computers for writing their stories. Written questionnaires were completed by the first-grade teacher to assess her attitude about using computers for classroom writing, as well as to document her own personal use of computers.

To describe a computer-enriched writing classroom, the study focused on four major questions: 1. How does a computer supported writing process environment evolve? 2. How are children's interests in writing with computers stimulated and maintained in a computer-supported writing environment? 3. How does a computer fit in a whole language classroom? and, 4. What behaviors does the computer elicit from the participants (teacher, children, computer consultant) within the classroom community?

Discussion and Implications

Data demonstrated that first-grade children can indeed write using a word processor. The computer changed the writing task by taking the focus off the physical process of letter formation and spacing, freeing children to concern themselves with communicating messages (Cochran-Smith, Kahn, & Paris, 1988). Word processing functions such as DELETE and SAVE were used by Room 14's first-grade writers to edit, to change their minds about the direction their stories were taking, and to add to their stories at a later time. The children I observed in this study did not seem to approach writing any differently whether they were at the computer or writing with pencil and paper. They did not seem to have a great deal of difficulty writing at the computer. Playing, however, seemed to be a pattern that emerged for most of the children when they were first introduced to the computer and continued for some throughout the study. Most children began to write by pretending to type or by hitting one key over and over. They seemed fascinated with the DELETE key and the space bar. Many children wanted to know how to make capital letters and would hold the SHIFT key down continuously while typing their name. They were quick to learn the rules needed to get by in a computer writing environment.

Writing is a complex process. How do children manage to write anything with the computer? The children in Room 14 did not seem to be aware of the complexities that Lynn, their teacher, and I were prepared to deal with. They appeared to write what they knew and ignored or worked around what they did not know. The literature and data drawn together in this study suggest that children should be allowed to play and experiment with the activity of writing, whether they are writing with pencil and paper or writing with a computer. This allows children to develop at their own pace. Most of the children in Room 14 did not have a computer at home, and those that did, did not use the computer for frequent writing. Since the computer has not been included in the common tools of writing found in most homes, formal schooling may be a first introduction to the computer for most children. A child's first step toward literacy may not look much like adult activities, but in this study, I observed children's playing and pretending channeled towards the adult activity of "writing so we can read it." Their playing and pretending may have been stifled by Lynn's lack of computer knowledge and discomfort at the computer. Because of her own developmental stage, Lynn focused on computer mechanics and consequently her orientation to instruction followed a more limited and structured approach.

One example of a first grader's playing and pretending was a scenario featuring Bruce. Bruce's first experience with the computer resembled play, as shown by the following sample of computer generated writing:

BRUCE TWENHAFEL
kjmgnfbkerjmdxorkldncmkdmkhsni
hrieruilepwi0pi-i0-pu-u49u0eu
e9w0u59ihfiehkh;ouo]phfoie
whsl;holi;hhueoihoypoteueohfoihoif
ilfhoiefefjophphfdhhfuhkhjg
hkjgkjgfhkgurghlkuyguioreuoiruyio
rouheguhrgeaukhoiruyortuhi
ogjkhluiooiurhhoigppoihrgeoihrl;yg
efoipfghioprgl;yfihoprgpohu
]rehou]roh6u]gohu]ieuopfsepi4kpow
euoperu]gipojoperh9erpoerol
qdjfroeugjoijrujikdjklfjfkjdlkaju
rogaijlkjoiruo]p8rojhrikjhf
idhgirjifkejirkejfodethjoehfwehew
hsprehioewrhoil;eholwsi;ioie
wrihoehoweholewioheihopwihoeoil;
hpewoihhehpoliewil;phohpoi
weoihptihpepoihiohief;rihihrohiop
hipoei;hsoliiohl;a wpihoioh;38io'1ier
iiqsda;whiohreifhol;awsoijioiuk
ojmxolyrgftjmdxolqfjrlgjmnrlf nmk
(Gunn, 1990)

To type his name, Bruce asked how to get capital letters and began using the SHIFT key. Then, with an impish grin, Bruce kept his fingers on the keys, pretending to touch type as he kept his eyes on those people around him who might be watching. He seemed to be imitating a typist who is able to type quickly and does not need to watch the screen. Bruce interrupted this make-believe typing to ask the question, "Does this keyboard have one of those letters that is a stick with a dot on it?" (Gunn, 1990). Bruce was making designs that looked like writing, but they were still a long way from the real thing. Lynn, Bruce's teacher, reacted by telling him that he was at the computer to write, not to play. By comparison, however, his writing with paper and pencil consisted primarily of illustrations with only a small amount of writing, usually to label his art work (see Figure 1).

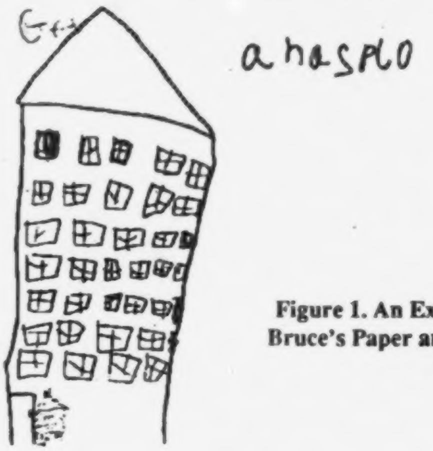


Figure 1. An Example of Bruce's Paper and Pencil Writing

Lynn's beliefs about how children learn to write directed her student's writing development at the computer. Children were expected to use Writer's Workshop time for writing stories, not playing.

Educators need to look at the whole environment rather than at individual children when deciding whether computers have a place in a classroom of beginning writers. The environment includes not only physical surroundings, such as the computers, children's books on chalk trays, and desks with paper and pencil. It also consists of human relationships. The relationships between the children and their teacher and among the children, should not be left out of the study of a writing environment. Looking just at a child's writing process, or even their writing product, gives a glimpse of what is going on, but from my observations of children in Room 14, it seems we must look not only at the child, but at the teacher and her assumptions and beliefs about literacy learning.

It was also evident from the study's findings that children's perceptions of what computers are used for might influence their own use of the computers. Generally, children in this study who viewed computers as a medium for playing games played with the word processor, while children who viewed computers as writing tools used the computer to write stories. Examples of the range in perceptions were seen most clearly with two children in Lynn's class. Drew and Morgan had apparent differences in perceptions of computers and their use, as evidenced by their behavior at the computer. Drew, probably the most proficient user of the keyboard, told me the computer in their home was used for "writing my homework when I'm sick and my mom uses it for her homework for school." Drew used the classroom computer only for writing, even during his first visit when others around him were experimenting and playing with *FrEdWriter*. (Rogers, 1986), a public domain word processor used for this study. An example of Drew's writing includes this story written with paper and pencil over several half-hour sessions:

Charles And Drew
By Drew
Charles wit srten [Charles went surfing*)
I wit srfn [I went surfing*)
We cipt on go [We kept on surfing*)
We stil cipt on goer [We still kept on surfing*)
We got to shoo [We got to shore*)
We wit hom [We went home*)
We LevD HaP ULee evraFrt [We lived happily ever after*)
The End (Gunn, 1990)

(*Child's oral interpretation)

Drew wrote quietly and surely at the computer, asking for little help as he searched for the keys that would spell his words on the screen. One of his first stories on the computer during a fifteen minute segment, while much shorter than his paper and pencil story above, continued in the same style. His writing was consistent, however, as Drew went from one writing medium to another:

Drew
I gt awt [I got out*)
I wit to the Stur [I went to the store*)
I withm [I went home*)
I Loiv hapu Leevtraftr. [I lived happily ever after*)
(Gunn, 1990)
(*Child's oral interpretation)

Using Drew's behavior at the computer as a guide, it was evident that a computer was a tool for writing, just as his paper and pencil were writing tools when he was at his desk.

In contrast, Morgan, Room 14's resident speller and writer of chapter books, did not begin to use the computer for writing until his sixth visit. He enjoyed using *FrEdWriter* as an arcade game, announcing to neighbors periodically what his score was by using the counter at the top of the screen. When Morgan was asked at the beginning of the study where he had seen computers outside schools and what they were used for, his reply included: "spaceships for taking pictures, fire stations where they type in problems, police cars for typing in problems, and little computers to shoot pictures into TV's." Morgan said computers were for "playing games," and when asked if he had any questions he would like answered about computers, he replied: "I'd like to know how they put the games inside the disk" (Gunn, 1990).

Morgan, a reader of Alfred Hitchcock stories, was working on a book with paper and pencil when I first began observing in Lynn's classroom at the end of December, and he continued with the same story until the end of February. While other first-graders were writing one sentence per page, Morgan's story was line after line of text on each page:

Night of The creapy huose
by Morgan

once there was a huose that creaked and creaked. But one dark dark Nite The door creaked open and nobody was at the door! And the poeple who lived there heard foot steps inside the house. they saw the bristles (changed to *curtains* in conference) But they kept on hearing more they wer very scared. [end of page 1]

But they kept on hearing rrrrrrrrrh! rrrrrh! creak! [end of page 2]

But they saw a little bit of skin and they found that it was there uncle. he was just waring his costume [end of page 3]

and they lived Happaly ever after. [end of page 4]

THE END [end of page 5] (Gunn, 1990)

A second story written with paper and pencil during this same period illustrates Morgan's knowledge of words, story structure, and spelling development:

Land Before Time
by Morgan

once apon a TiME There Wose a Grate earthquake

The Land SPLiT
Little foot and hiss MoTher Were seperated.

He Had To Walk the Grate vally By Him Self But Preetty soon sara Arived The Same Persen Little foot Saw at the begining.

for Days and Days They Walked But Then Ducky peatree and spike came to joun [join.)

But Then! Sharp Tooth Came! But then Little Foot Got an idea you Go Down Said Little foot To sara and get Behind him and make him turn eruond [around] to Get his atenckin [attention] and will [we'll] rock on him and heel [he'll] fall in the lake

They had lots of Tunels [tunnels] To go Threw [through] But! one Day way under They saw the great valley.

It was the Great vally!

The End (Gunn, 1990)

Morgan wrote only one story during his computer sessions:

MORGAN
THOMAS WILEY
The MOUSE AND THE CAT by Morgan
Once there was a cat and a mouse.
the cat always chased the mouse.
one day the mouse said I,am tired of
being chased all the time. help!
said the mouse. the cat is chasing
me. he ran into his hole and sat
there. he sat there for days and days.
but one night the cat went to sleep
and the mouse crept out of his hole. (Gunn, 1990)

Morgan was beginning to use the computer for writing, but he continued to write elaborate chapter stories at his desk. In late April, when I asked him what he was writing in a rough draft booklet, Morgan told me: "I'm writing a book on how to draw and it's hard to think of a way to start—there are three sections and all are different." (Gunn, 1990). From observing Drew and Morgan writing at their desk and at the computer, it appears that their perceptions of what a computer is used for guided their use of the computer for writing. When instructing young children to use a computer for writing, then, it would be beneficial to include their perceptions of the role of the computer into the instructional design, through teacher modeling and discussion of the computer as a writing tool.

Introducing and using computers and word processors in elementary school writing programs requires more than a global statement that "children and teachers benefit from such a program." Each member of Room 14's developing writing environment had a personal version of how things worked in this particular group, this educational microculture that was formed when computers were introduced into Writer's Workshop. Although this study describes one teacher's process and one classroom of writers, it reveals the complexity of technological change and school culture. To understand this diversity, it is helpful to view the classroom community, the values of the individuals participating in the classroom, the assumptions and perceptions of the individual teacher in relation to effective teaching and how children learn. Added to this is the need to look at language and literacy acquisition and development and the needs and abilities of the children participating in the classroom community. The literature reviewed and the findings reported in this study brought to light a key issue pertinent to computers in writing: staff development.

Staff Development

Adding computers to Lynn Tenille's existing writing program allowed me to see one teacher's struggle to learn to use a computer for writing at the same time her student's were using the computer in their Writer's Workshop. Lynn's struggle, as well as the small amount of curricular change noted during this six-month study, suggests that computer staff development is critical when a novice computer-using teacher introduces computers to children for writing.

This research suggests implications for school computer policy, especially in the areas of staff development and the integration of computers into existing curricula. The usual pattern is that schools purchase computers and make them available to classroom teachers. Teachers attend computer workshops and then are required to use the computer in their classroom. It has been suggested that appropriations for educational technology be divided equally into thirds among hardware, software, and staff development (Layton, 1990). Unfortunately, staff development typically receives comparatively little of the monies appropriated for technological advances. Without appropriate staff development, teachers are often uncertain about how to use a computer for writing themselves, thus, they are unable to introduce the computer to their students as a writing tool.

Teachers should have a working knowledge of a word processor before considering whether word processing fits with existing belief systems and goals for children's writing development. In this study, it was evident that Lynn's attempts to implement computers in her writing program were impeded by her inexperience with the computer. Word processing must be learned thoroughly by a teacher before it can be introduced to her students.

Teachers need time and opportunities to incorporate word processing into their classroom and into their curricula. The time and opportunities for such learning in this study were minimal for a practicing teacher. It appears that appropriate training in the use of computers for writing ought to occur at the preservice level, when students are preparing to become teachers. Are teacher training institutions adequately preparing our future teachers for integrating computers into existing curricula, or is computer education seen as a separate program? It is my suggestion that teacher training institutions view computer use in an integrated fashion within existing methods courses, where future educators see the computer as a tool and are made aware of the impact their own assumptions and beliefs have on the computer's use in classrooms.

Staff Development Model: Integrating Computers Into a Writing Program

As this study progressed, patterns emerged that have implications for staff development training. Information from my review of literature combined with findings from this study suggest changes in existing computer inservice. Other educators and researchers have suggested staff development models for education (Gall & Renchler, 1985; Joyce & Showers, 1988). Moursund (1989) has outlined a model for integrating the computer as a tool into curriculum. Based on my observations in this study, however, it seems that additional considerations are needed to integrate a computer into a writing program.

Ten staff development components are proposed from the culmination of this study. Admittedly, these components are hardly original, but taken together with qualitative information from this study, we must continue to emphasize the necessity for including each in a staff development model. Each component will be briefly presented with a discussion supporting its inclusion in this proposed and developing holistic staff development model for computer inservice.

1. Commitment to Children: Computers and their use in support of classroom writing demand that teachers continually keep their children's best interests in the forefront of their curriculum planning. The purpose of adding computers to a classroom writing program is to benefit the students and their teacher. If an activity can be done more effectively without the computer, then a computer should not be involved. Teachers need to ask, "Why am I doing what I am doing?" (Durkin, 1989) before adding a computer to their instruction. The computer should not become an overriding feature in classroom instruction, but rather an effective teaching and learning tool that benefits children. Computer inservice must address decision-making and its effects on children so that teachers learning to use the computer don't lose sight of their original goals for using a computer and word processor in their curriculum.

2. Commitment and Recommitment to New Technology and Change: Teachers involved in computer inservice must be committed to learning about technology. They must be willing to devote time and energy to learn to use the computer for their own writing, and this willingness must come from within. Directives from administration to learn and use a computer for instructional purposes are ineffective without the cooperation of the practicing teacher. Learning to use a computer for writing cannot be imposed. Furthermore, not all individuals are comfortable using the computer, even if they originally committed to change. A computer is a machine that can cause frustration and even fear. It is important to consider teacher aptitudes and individual learning styles, and to be prepared to slow down the pace when frustration and fear are recognized. Timelines should be flexible to account for recognizable development differences in individuals.

   Time and shifting beliefs may change a teacher's original commitment to learning about a computer and using it with children for writing. It is too easy to let time break down the original commitment as circumstances change, energy levels ebb, and new interests intervene. A recommitment is often necessary and this might involve restatement of goals or discussions regarding shifts in focus or concerns.

3. Knowledge of Word Processor: A teacher must be able to use a word processor before working with children at a computer. Learning to use a word processor can be a long, frustrating process for some individuals; however, as in learning any new skill, time, effort, and patience will usually overcome problems. Learning to use the computer at the same time one is teaching it to children is an invitation for frustration and possible failure. Administrators and teachers must understand the need for time—learning to use a word processor efficiently takes more time than is usually allotted for computer staff development.

4. Computer Consultant Support: Teachers need long-term help and support if they are to use computers and integrate word processing into their writing curriculum. This support comes in several different ways. It seems critical that someone knowledgeable in computers, staff development, and in how children learn be available to work with teachers once they know word processing themselves. A practicing teacher does not have time to troubleshoot when a printer does not print a child's story or when vague messages appear on the screen when a child or teacher is in the process of writing and wants to save a story for future writing. Given these frustrations, the computer and its use as a writing tool are cast aside by many teachers as too much bother. A third party, either a computer consultant who visits a school regularly or another teacher who is given time to work with staff, seems necessary to help teachers through the rough spots. It is not sufficient to

designate a computer-knowledgeable teacher actively teaching in a classroom as the expert for others to consult.
It seems advisable to have a computer consultant work with small groups of teachers periodically (e.g., once a week in the earliest stages of the staff development process, once a month as teachers are more familiar with the word processor or begin using the computer in their classrooms). Teachers can meet during these inservice times to present questions, engage in discussions pertinent to their own teaching situations, problem solve, and practice using the word processor. This should also be a time for goal setting and reaffirming commitments.

5. Demonstrations and Modeling: Demonstrations of children writing with computers are important to goal setting. Teachers must see the possibilities and benefits available to them once they have mastered the word processor themselves. These demonstrations need to occur periodically. At first, demonstrations should be used to show possibilities of computers for writing. Later, demonstrations can be useful to stimulate ideas once word processing is mastered and teachers are ready to use a computer for writing in their classroom.

6. Goal Setting and Resetting: Learning to use a word processor demands a context. Personal writing such as letters, lesson plans, memoranda, or worksheets are an appropriate context for an individual learning to use the computer for writing. Later, a word processor can be used for children's story writing, group writing activities, or modeling reading and writing strategies. Not all teachers will want to use the computer for writing in the same way, and it is important that teachers set their own goals so they can visualize computer use, as well as guide the direction of their staff development. These goals should be reviewed periodically as teachers gain skills.

7. Peer Group Support and Collaboration: Beginning a new program can be a lonely and frustrating experience for a sole teacher implementing a new curricular component. Teachers working together, sharing failures and successes at comparable stages of learning, seems to be a necessary component for effective implementation of curricular changes.

8. Individualized Instruction: Just as children develop at individual rates, so too do adults. Teachers need individualized instruction on the computer within a group setting of their peers if frustration is kept to a minimum. Instructing by grade level is helpful, inasmuch as teachers and the instructor are able to personalize information and instruction. Flexibility is important so that each teacher's needs are met.

9. Recognition and Reaffirmation of Beliefs: It is legitimate to anticipate that a shifting of focus or new interests might obliterate a teacher's original ideas or beliefs (e.g., in how children learn, how computers can or should be used in the classroom). A necessary component of computer inservice is to help teachers become aware of the impact that their own assumptions and beliefs have on the computer's use in classrooms. A reaffirmation of beliefs might include directed discussion or development of new goals. In Lynn's case, an explicit recognition of her beliefs in how children learn to read and write might have changed her expectations of children using the computer for writing.

10. Time: Time should be addressed in two ways through this staff development model. First, time must be viewed as a long-term commitment: novice computer users can not usually learn to use a computer for their own writing and introduce it to children in a short amount of time. School districts must see staff development as on-going. A proposed inservice plan might consider one year for teachers to learn to use the computer with a second year to implement computer use in the writing program. A third year might be appropriate for follow-up and for integration of the computer and writing into other content areas.
Secondly, teachers should be given the time and opportunities for their own computer learning through released time. Teaching combined with self-learning is an exhausting and frustrating experience. Giving teacher plenty of time to explore, to play, to take risks, and to get to know the computer is an important component of staff development. One successful method used by an elementary school in Hillsboro, Oregon, included hiring three substitutes for one day to rotate from one grade level to another. Each grade level (three teachers) signed up for a designated release period, and the substitutes rotated throughout the building. This solution allowed teachers to concentrate on using the computer during a designated time with a consultant available without being pulled back into their teacher role. The collegial issue is addressed with this method as well, as teacher in one grade level or content area are able to share their experiences, fears, processes, and finished products with each other.

These ten components are not necessarily linear or in any set order, but are interactive, ongoing, and recursive. Just as background knowledge, individual rates of readiness, and styles of learning provide important guidelines for planning instruction for children, so too should they be considered a framework in planning staff development for adults.

Conclusion

Is there a place for computers in first-grade whole language classrooms? Evidence from this study suggest that they are appropriate when supported by the teacher's and children's beliefs and practices. It is important for teachers to select the computer use that reflects and supports their educational goals and facilitates teaching and learning. But first, teachers must be provided appropriate staff development and time that will allow them to learn to use the computer themselves.

References

Cochran-Smith, M., Kahn, J., & Paris, C. (1988). When word processors come into the classroom. In J. Hoot & S. Silvern (Eds.), *Writing with computers in the early grades* (pp. 43-74). New York: Teachers College Press.

Daiute, C. (1986, May). Physical and cognitive factors in revising: Insights from studies with computers. *Research in the teaching of English, 20* (2). 141-159.

Durkin, D. (1989). *Teaching them to read*, (5th Ed.). Needham Heights, MA: Allyn & Bacon.

Gall, M., & Renchler, R. (1985). *Effective staff development for teachers*. University of Oregon, Center for Educational Policy and Management.

Gunn, C. (1990). A descriptive study of a developing computer supported writing environment in a first grade whole language classroom. Unpublished doctoral dissertation, University of Oregon, Eugene.

Hawisher, G. (1989). Research and recommendations for computers and composition. In G. Hawisher & C. Selfe (Eds.), *Critical perspectives on computers and composition instruction* (pp. 44-69). New York: Teachers College Press.

Herrmann, A. (1989). Computers in public schools: Are we being realistic? In G. Hawisher & C. Selfe (Eds.), *Critical perspectives on computers and composition instruction* (pp. 109-125). New York: Teachers College Press.

Hoot, J., & Silvern, S. (1988). Word processing as a tool for developing young writers: An overview. In J. Hoot & S. Silvern (Eds.), *Writing with computers in the early grades* (pp. 3-9). New York: Teachers College Press.

Joyce, B., & Showers, B. (1988). *Student achievement through staff development*. New York: Longman.

Kahn, J. (1987). Learning to write with a new tool: Young children and word processing. *The Computing teacher, 14* (9), 11-12, 56.

Moursund, D. (1989, April). Change processes in education: What does the literature tell us? In S. Franklin & N. Strudler (Eds.), *Effective inservice for integrating computer-as-tool into the curriculum* (pp. 1-8). Eugene, OR: International Council for Computers in Education.

Rogers, A. (1986). *FrEdWriter* [Computer Program]. San Diego: Computer Using Educators Softswap Project.

Routman, R. (1988). *Transitions: From literature to literacy*. Portsmouth, NH: Heinemann.

Simonson, M., & Thompson, A. (1990). *Educational computing foundations*. Columbus, OH: Merrill.

Stonier, T., & Conlin, C. (1985). *The three C's: Children, computers and communication*. Chichester, England: John Wiley & Sons.

Projects
Teacher Training

Teacher Explorer Centers: Introducing Michigan Teachers to MultiMedia

Anne Porter, Fred D'Ignazio
Oakland University
Rochester Hills, Michigan

Abstract

In Michigan, the State Board of Education, a K-12 school district, a community college and a university teacher training program have joined with business partners to introduce Michigan's teachers to educational applications of multimedia technologies through Teacher Explorer Centers.

A Teacher Explorer Center is a place where classroom teachers gather in teams of 3 teachers and 1 administrator to explore classroom applications of the integration of microcomputers, interactive video disks, CD ROMs, VCRs, camcorders, hypermedia and telecommunications technologies. Each team spends six hours, hands on, in a classroom equipped with state-of-the-art equipment furnished by business partners. Their task is to create their own multimedia production, much as their students might some day in the near future.

At the end of the first year of the project (October, 1990), over 4000 teachers had applied to participate in Teacher Explorer Center Workshops. In response to this overwhelming demand, Michigan's State Board of Education expanded the funding of the first year's single site and added two more Teacher Explorer Center training sites for 1990-91, the second year of the project.

This project presentation will describe the goals, implementation structures and outcomes of the exceedingly successful project.

Implementing a Staff Development Program in a Large School District

Sharon Bell
New Orleans, Louisiana

Abstract

Abstract not available at press time.

Total Integration of Computers/Multimedia Technology into a K-12 Curriculum

Tony Kietrys
Jack Kmetz
Wallenpaupack Area School District
Wallenpaupack, Pennsylvania

Abstract

The objective of this presentation is to offer solutions to other school districts who are or may be planning to integrate computers and multimedia technology into their districts. The presenters plan to hold an open forum for discussion which will include the positive outcomes along with the problems encountered during the 6-year plan. At the conclusion of the project presentation, each participant will receive handouts detailing the long range plan. Special consideration will be given to the cost of implementation along with the actual breakdown by building and class level of the hardware and software used, and how it relates to the curriculum being taught.

The Wallenpaupack Area School District is a rural school district in Northeastern Pennsylvania. It services approximately 2,000 students in grades K-12. The district is currently in the 6th year of a long range computer plan. The goal of the plan is for total integration of computers and multimedia technology into the curriculum.

Paper

The Development of an Evaluation Handbook For Computer Integrated Instruction

Richard Mowe
St. Cloud State University , Minnesota

Abstract

A research and development (R & D) design was used to produce a program evaluation handbook for the evaluation of computer integrated instruction (CII) at the elementary school level. The handbook was reviewed by a panel of 13 experts in computers in education and experts in program evaluation. The experts responded to a questionnaire and commented directly in the handbook to give helpful feedback and suggestions for improvement. The handbook was then revised and field-tested at 10 sites. Users reviewed the handbook, completed a fieldtest questionnaire, and filled-out a time log of evaluation activities. Then they used the handbook to review their school or classroom CII programs. Among the findings of the study was the conclusion that the handbook was very easy to use. The fieldtest process took an average 7.63 hours. The handbook should be revised to aid in summarizing and reporting questionnaire data and to provide instructions for modifying questionnaires.

The handbook assisted in gathering information and reporting the results of the school's CII program. It was more helpful for gathering information than reporting results. The handbook helped users gain an accurate description of their CII program and gave them a secure basis for revising their program. On the basis of the conclusions, recommendations were made to revise the handbook and conduct one more fieldtest.

Introduction

As the availability of computers in schools has increased rapidly in recent years, computer integrated instruction (CII) has become commonplace. But few educators evaluate the effectiveness of CII in their schools or classrooms. The goal of this study was to develop a formative program evaluation handbook for elementary CII programs enabling educators with little or no formal program evaluation training to (a) gather and interpret program information and (b) make recommendations for program improvement based on that information.

CII is "the integration of computer-as-tool into the school curriculum" (Moursund, 1985, section 8, p. 5). CII forms the object of program evaluation for the handbook produced in this study. The book also facilitates the evaluation of keyboarding, problem solving, and simulation software, which are closely related to CII.

For the purpose of this study, evaluation means formative evaluation, which is a process that involves:

1. Determining information to collect.
2. Collecting the information.
3. Describing the program.
4. Recommending program improvements.
5. Reporting the results.

Handbook Summary

The evaluation process is a four step method for obtaining information regarding the elementary school CII program: planning the evaluation, collecting the information, analyzing the data, and reporting the results. The process is aided by a number of questionnaires in the handbook.

Planning the Evaluation

The first step is to decide what information to collect. This will depend on the stage of development of the CII program. For instance, a school that has implemented a CII plan will require different evaluation information than a school that has not written a CII plan.

The next step is to decide who will gather the data. One person familiar with the program may gather the data or several people could divide the responsibilities. A schedule for data collection and interpretation is also helpful.

Collecting the Information

The handbook organizes information collecting via questionnaires into the following areas: program description, goals, staff computer knowledge and skills, plan, implementation and goal attainment, and general strengths and weaknesses.

The program description assesses the current state of CII in the school. To make recommendations for improvement, it is necessary to study the current program to understand what needs changing. The first section of the program description deals with the context. It provides background information on the school, community, district, financial outlook, and unique problems and opportunities with which the CII program interfaces.

The next section asks for distinctive computer uses within the school. Then the administration of the CII program is examined. In the instructional delivery section, forms are provided for entering computer use by grade and a hardware inventory.

The next questionnaire assesses the CII goals. Participants are asked to enter the goals and then examine them. They enter the priority of the goal and how goal attainment can be verified. They are asked if there is sufficient hardware and software and if teachers are adequately trained.

The staff questionnaire is divided into two parts. The first part assesses the skill level at using the computer. The second part asks for the types of computer use employed by the staff in the classroom.

The plan questionnaire examines the types of activities that are used to support the CII goals. Activities that support the goal are listed along with further training and resources required.

The activity implementation/goal attainment questionnaire would be useful in CII programs where CII plans have already been implemented. For each goal, the targeted grades and evidence of attainment are listed. Then, skills taught can be entered along with the percentage of the class attaining the goal and further resources needed.

In the last questionnaire, strengths and weaknesses of the school CII program are examined independently of goals. The purpose of this questionnaire is to gauge participant satisfaction and gather suggestions for improvement.

Analyzing the Data

After the data is collected, it is compiled by combining questionnaire responses into a composite. Then, conclusions are drawn and recommendations are suggested.

Reporting the Results

Finally the results are presented in written form. Background information is first given to explain why the evaluation was instituted. Next, a description of the current program is given. Then, evaluation questions are posed and discussed. Finally, recommendations are presented.

Research Summary

Literature

The literature was examined with the following questions in mind: what should be evaluated and what are the standards of evaluation?

Past studies of computer assisted instruction (CAI) projects have measured student achievement on tests (e.g., Kulik & Kulik, 1987; Niemiec & Walberg, 1985). The standard of comparison for students taught by CAI was test scores for students taught by a teacher.

Billings (1985) developed a handbook for evaluating computer education programs in which she suggested that the evaluation should examine four aspects of the computer education program: objectives, teacher training, materials, and student learning. The standard of comparison was performance in relation to stated goals.

Goal attainment can not always be measured. In cases where goal attainment cannot be measured directly, Berman, Greenwood, McLaughlin, and Pincus (1975) suggest using perceived goal attainment as the standard. Another standard they suggested is a change in behavior. Fink and Kosecoff (1978) suggest testimony from a student or teacher that learning has occurred as a standard.

Many computer education programs are not at the point of evaluating student performance in relation to stated goals. Concerns are writing goals or monitoring activities that may lead to goal attainment later. Checklists provide ready-made data collection instruments for monitoring computer use and aiding in planning for school or district computer use (Coe, 1985). The Wisconsin Department of Public Instruction (Anderson & Shires, 1987) checklist collects information in the areas of the district plan, district organization/implementation, staff development, computer hardware, software, computer facilities, and curricular integration.

Other program evaluation information may be found in computer curriculum planning books (for instance, Anderson, & Shires, 1987; Hawaii Department of Education, 1987; Merrimack Education Center, 1984). A strength of the book sections is that they put evaluation within the broader context of planning for computer education programs.

Evaluation can also take place in the absence of goals (House, 1973; Scriven, 1974). One standard for success of a program is the satisfaction of participants. Another is practicality. For instance, is the program affordable?

According to the literature, some of the dimensions of CII that an evaluation handbook might address are use of software, teaching strategies, and curriculum concerns. The objects of evaluation mentioned can be classified as plan-related, teacher training, materials (including hardware, software, and printed materials), and student learning. Other areas that are important to be evaluated, but not mentioned, are a program description and general strengths and weaknesses of the program. Standards of comparison could be goal attainment, perceived goal attainment and criteria not related to goals.

Method

A research and development (R & D) design was used in this study to produce a program evaluation handbook for the evaluation of computer integrated instruction (CII) at the elementary level. These steps were followed: (a) Study the literature, (b) write an initial version, (c) review and revise the initial version, (d) fieldtest the second version, and (e) make recommendations for a final version (Billings, 1986; Borg & Gall, 1983).

The Writing and Review Process. The handbook was designed for internal evaluators operating at the elementary school level. Possible evaluators could be principals, teachers, or specialists working alone or as members of an evaluation team. The handbook was written to be as short as possible and designed to be used as quickly as possible. Since designing data collection procedures can be a time consuming part of an evaluation, the handbook would use pre-made questionnaires. The focus of the handbook was decided to be formative evaluation, which also includes needs analysis, an implementation description, and goal attainment measurement. It was assumed that schools using the handbook would be supporters of or interested in starting CII programs. The components evaluated were to be the program description, goals, planning, teacher training, student learning, and general strengths and weaknesses of the program. The standards for evaluation could be goal attainment, perceived goal attainment, or criteria unrelated to goals.

The handbook was designed to facilitate the following steps of evaluation: (a) planning the evaluation, (b) collecting the information, (c) interpreting the information, (d) and reporting the results.

The following research questions were used in both the review and fieldtest processes:

1. How easy is the handbook to use?
2. How long does the evaluation take?
3. How should the handbook be revised?
4. To what extent does the handbook assist the computer educator in gathering information and reporting the results of the school's computer integrated instruction program?
5. To what extent do schools that use the evaluation handbook have an accurate description of their computer integrated instruction program?
6. To what extent do schools that use the handbook have a secure basis for making decisions for revising their computer integrated instruction programs?

For the review process, the questions were asked from the point of view of someone who had not actually used the handbook, but was predicting the results of handbook use.

An expert panel reviewed the initial version of the handbook by reading and commenting in it and completing a questionnaire. The panel was composed of representatives from the general categories: (a) computers in education experts, (b) program evaluation experts, and (c) those participating in or affected by computers in education program evaluations. Most of the panel members qualified in more than one category.

This study used review and fieldtest questionnaires modified from the Billings (1986) study. The questionnaire featured both closed- and open-ended questions. The closed-ended questions could be answered (yes/no) or (1-4) where 1 = *Excellent* , 2 = *Good*, 3 = *Fair*, and 4 = *Poor*. Since the main purpose of the questionnaire was to show which parts of the handbook needed revision, the questionnaire responses were recoded to reflect that emphasis. Responses 1 and 2 were combined to signify *needs no revision*, and 3 and 4 combined into *needs revision*.

After each question, open-ended comments were invited. The questionnaire covered handbook goals and objectives, handbook use, handbook results, and recommendations.

The closed-ended responses were tallied with frequency counts. The open-ended comments were compiled into a master list .

The Fieldtest Process. On the basis of the review process the handbook was revised and then sent to schools for fieldtests. Fieldtest personnel had a twofold job: reviewing the handbook and actually using the handbook to review the school or classroom CII program.

The handbook was tested at 12 sites. Educators used the handbook for three weeks to evaluate their school or classroom CII program. The study categorized schools by population of the town or city where the school is located and degree of CII implementation. The population categories were small, medium, and large, where small = 7,000 or less, medium = 7,001 - 100,000, and large = 100,001 or more. The CII implementation categories were *much CII implementation* and *little CII implementation.*

As fieldtest personnel reviewed the handbook, they answered the research questions, this time from the perspective of actual handbook users.

The fieldtest employed two measures: the fieldtest questionnaire and the time log. The questionnaire had many questions parallel to questions in the review questionnaire. Participants were asked to answer yes/no or 1-4 and make comments after each response.

The purpose of the time log was to record evaluation activities completed and the time spent. Participants could enter the code for prespecified activities, or could enter their own categories of activities. The prespecified activities were *read the handbook*, *planning*, *administer questionnaire #__*, *interpret the data*, or *report the results*.

The closed-ended responses were tallied with frequency counts. The open-ended comments were compiled into a master list. The times taken to complete evaluation tasks were averaged. Total time was reported, as well as the time spent for various categories.

Results

The testing of the handbook took place in two phases. The first phase was the review phase in which a panel of experts reviewed, but did not use the handbook. The second phase was the fieldtest phase, in which elementary school educators evaluated their school or classroom CII programs and evaluated the handbook. A summary of the field test results is discussed here. Detailed review and field test data are presented in Mowe (1989).

The conclusions presented emerge from the fieldtest questionnaires and the handbooks themselves. Answers to the research questions are given, along with suggestions for a future handbook revision.

Question 1: How Easy is the Handbook to Use? In general, the handbook is very easy to use. A clear majority of responses showed no need for handbook revision. Two comments suggested changes for the handbook. One was to provide compilation forms to help analyze the data and report information. Another was to provide a goal attainment form that the school or district could adapt to its own uses.

Question 2: How Long Does the Evaluation Take? Four time logs were submitted. The average time for the evaluation (including completing the fieldtest questionnaire) was 7.63 hours. The approximate average time for completing the individual questionnaires was one hour each.

The return rate for time logs was 40% overall and 10% for members of the *much CII implementation* group. It is assumed that the low return rate was related to the pressure to complete the project within the allotted time. The participants not turning in time logs did read and reviewed the handbook and completed the fieldtest questionnaire, but did not complete the time log and the evaluation of the school CII program. It is further assumed that those in the *much CII implementation* group had greater time pressures because they had to evaluate a larger program. Thus they had an even lower time log return rate.

Therefore, if the fieldtest had lasted longer, the average may have been higher. More research is needed to determine the length of time needed to use the handbook.

Question 3: How Should the Handbook be Revised? Quantitative ratings revealed no cases of needed handbook revision. Suggestions for revision emerged from an analysis of the handbooks and the qualitative comments associated with this and other questions. The recommendations are summarized:

1. Provide compilation forms for the analysis of data and reporting of results.
2. Shorten the evaluation process.
3. Include sample forms and directions for modifying them.
4. Provide the means for evaluating attitudes.

Question 4: To What Extent Does the Handbook Assist the Computer Educator in Gathering Information and Reporting the Results of the School's CII Program? In all areas related to this question, participants rated the handbook as *needing no revision.* An analysis of the handbooks turned in, however, revealed that not all the questionnaires in the handbooks were used.

The majority of participants completed the following questionnaires:

- Program Description Questionnaire
- Goal Questionnaire
- Staff Questionnaire
- Strengths and Weaknesses Questionnaire

Only one of the field-testers submitted an evaluation report. The following questionnaires were hardly used:

- Planning Questionnaire
- Plan Questionnaire

The handbook appears to be more helpful in gathering information than reporting results. This could be attributed to a lack of time or to the intrinsic difficulty of interpreting and reporting results. If the problem is too little time, then handbook users can solve the problem by using the handbook as long as needed. If the time is related to the difficulty of the task, then the handbook needs to be revised to aid in producing reports.

Question 5: To What Extent Do the Schools That Use the Evaluation Handbook Have an Accurate Description of Their CII Programs? According to field-testers, the handbook provided an accurate description of a school CII program.

Question 6: To What Extent Do Schools That Use the Handbook Have a Secure Basis for Making Decisions for Revising Their CII Programs? Field-testers felt that the handbook was very helpful in providing a secure basis for revising the school CII program.

Summary of Conclusions

A summary of the answers to the above questions reveals the conclusions of this study:

1. The handbook is very easy to use.
2. The evaluation and handbook review process took 7.63 hours. This time should be viewed as a minimum time.
3. The handbook should be revised to aid in summarizing questionnaire data and to provide customized questionnaires for districts.
4. The handbook assists in gathering information and reporting the results of the school's CII program. It is more helpful for gathering information than reporting results.
5. Handbook users felt that the handbook helped them gain an accurate description of their CII program and a secure basis for revising the program.

Recommendations

On the basis of the conclusions, the following recommendations are offered:

1. Revise the handbook.
2. Conduct one more fieldtest.

The following revisions would be helpful:

A. Provide compilation forms for the analysis of data and reporting of results.
B. Include sample forms and directions for modifying them.

Handbook users completed questionnaires as part of the evaluation of their CII programs. But once they had completed the questionnaires, they needed a way to summarize data and report results from questionnaires. The handbook gives instructions for the compilation, but more is needed.

Two possibilities exist. First, information could be entered into a matrix. Then, the composite information could be compiled with the aid of a calculator. Another possibility is to use an *AppleWorks*[1] or *Microsoft Works*[2] spreadsheet template to directly perform the necessary calculations. The word processing capabilities of *AppleWorks* or *Microsoft Works* could then be used to make up a report template, which could be combined with the data from the spreadsheet.

One of the comments made was that the handbook needs to present sample forms and provide advice for modifying those forms for the unique needs of schools. One way to do that would be to publish the handbook on disk as an *AppleWorks* or *Microsoft Works* word processing file.

An added fieldtest would have two emphases. First would be more accurate time estimates for the parts of the evaluation process. Second would be an examination of the assistance rendered by the handbook in reporting the results of an evaluation.

To encourage participants to complete time logs and evaluation reports, the time of the fieldtest should be lengthened from three to six weeks. Also, fieldtest personnel could be reminded to submit time logs and written evaluation reports.

References

Anderson, M., & Shires, L. (1987). *A guide to curriculum planning in computer education*. Madison, WI: Wisconsin Department of Public Instruction.

Berman, P., Greenwood, P.W., McLaughlin, M.W., & Pincus, J. (1975). *Federal programs supporting educational change: Vol. 4 (abridged), A summary of the findings in review*. Santa Monica, CA: The Rand Corporation.

Billings, K.J. (1986). *The development of an evaluation handbook for a computer education program*. Unpublished doctoral dissertation, Columbia University.

Billings, K.J. (1985). *An evaluation handbook for a computer education program*. Eugene, OR: International Council for Computers in Education.

Borg, W.R., & Gall, M.D. (1983). *Educational research: An introduction (4th edition)*. New York: Longman.

Coe, Merilyn, (1985). *District computer concerns: Checklist for monitoring instructional use of computers*. Portland, OR: Northwest Regional Educational Laboratory. (ERIC Document Reproduction Service No. ED 270 088)

Fink, A., & Kosecoff, J. (1978). *An evaluation primer*. Beverly Hills, CA: Sage.

Hawaii Department of Education. (1987). *Department of Education Computers in Instruction Plan*. Hawaii: Author.

House, E.R. (Ed.). (1973). *School evaluation: The politics and process*. Berkeley, CA: McCutchan.

Johnson, V. (1988). *An exploratory case study describing the long term residual effect of the computer integrated instruction inservice (CI3) project*. Unpublished doctoral dissertation, University of Oregon.

Kulik, J., & Kulik, C. (1987). Review of recent research literature on computer based instruction. *Contemporary Educational Psychology*, 12, 222-230.

Merrimack Education Center. (1984). *Applications planning: A guide to planning and implementing a district-wide computer program*. Chelmsford, MA: Author.

Moursund, D. (1985). *High tech/High touch: A computer education leadership development workshop*. Eugene, OR: Author.

Mowe, R. (1989). *The development of an evaluation handbook for computer integrated instruction*. Unpublished doctoral dissertation, University of Oregon, Eugene.

Niemiec, R.P., & Wallberg, H.J. (1985). Computers and achievement in the elementary schools. *Journal of Educational Computing Research*, 1(4),435-440.

Scriven, M. (1974). Evaluation perspectives and procedures. In W.J. Popham (Ed.). *Evaluation in education* (pp. 30-93). Berkeley, CA: McCutchan.

---

[1] CLARIS Corporation; P.O. Box 526; Santa Clara, CA 95052

[2] Microsoft Corporation; One Microsoft Way; Redmond, WA 98052-6399

Paper
School Administrators and the Management of Technological Change

Kieth C. Wright
University of North Carolina-Greensboro

Introduction: The Technological Revolution

The economic and human revolution of technology comes into our communities, our homes, and our schools. This revolution could increase school administrators' power to manage information more effectively in the "information age" and to be a more powerful player in that age. Such information management skills will clearly be needed as we move toward site-based management in our schools. But what about the reality? This paper addresses the possibility of using the power of computer-based technology in our school systems rather than being used by that technology. The author's suggestions regarding specific software applications have all be evaluated in the context of both K-12 education and higher education while consulting with school administrators. Effective school information management is possible. The major obstacle to effective school utilization of technology is school personnel resistance to change. Helping school personnel deal with change is the focus of this paper.

Defining the Impact of Technological Change on School Administration

Computer and communications technology are rapidly altering the society in which schools and school systems operate. More and more information about business and services is available via computer. Businesses, consumers, researchers, and citizens expect to have access to current information as close as their telephone. Steward Brand, in his The Media Lab (1987), states that our ability to digitize information of all types is as important as the invention of the movable type printing press. The ability to communicate and transform information into a variety of formats make all information more flexible. A much wider variety of published products can emerge and the form of information tends to remain flexible (and modifiable) over time.

Seiler (1989) notes that a single format: the digital electronic medium becomes possible for printed text, graphics, photographs, film, video and manuscripts. She lists five technologies which make the ways we deal with information radically different:

- computer - providing a tool for rapid information manipulation and storage
- communication - providing for the rapid transfer of many different formats over long distances at high speed
- databases - providing access to full text and graphics of information resources
- sensors - providing a rapid means of converting text and graphics to machine readable form with scanners, video images, voice recognition
- multimedia programs - providing easily used programs for expert software systems, voice recognition, speech synthesis, imaging, etc.

We need to stress that communication by computer is not the same as face-to-face communication. Information is communicated face-to-face and via computer that can not be communicated the other way. The power structures of schools are affected by these differences. In a computer communication system (or network) there is less filtering of information within the system. Negative information (parent's complaints, grade reports, delays in delivery, cost overruns) can no longer be hidden or delayed. Wright's law which states, "negative information does not flow uphill because that's where to money comes from" can be overcome if the computer network allows school administrators to have access to information typically routed "through the system."

On the other end of the system, the school employee with a computer can have access to cost figures, enrollment projections, and working documents (unless they are "secured" in the system). Both data and interpretation of "how things are going" is much more widely spread throughout the system. Such open systems are effective in horizontally managed organizations which require information and input from a variety of sources. Schools are likely to be traditional hierarchical organizations and are not so accepting of open access to information throughout the system.

Scientific American (1989, p. w1) introduces the special section on "the workstation revolution" with these words:

> Every year new stories about the emergence of faster, more functional, and, in defiance of generations of economists, cheaper computers appear. And every year come new claims that we are in the midst of a revolution. We are in the midst of a revolution. It is not simply because computers are getting smaller and cheaper. The word revolution implies something more dramatic. The computer

> revolution is about the fundamental change occurring in the relationship between human beings and computers. The way we work with computers is changing dramatically. In this revolution, the workstation has become the most visible agent of change.

The Power of School Administrators Meets the Power of Technology

State and local government agencies, accrediting agencies, and state departments of education have increasingly required the use of computer-based information systems. The systems are upon us; the question remains can school administrators use the power of these systems to empower their schools? Or, will these new systems simply create more demands on administrators, teachers and students — thereby weakening all personnel and the entire system? Kinnaman (1990, 43) stress that...

> Although computer power will continue to increase at astounding rates, technology is still just a vehicle. It is not a destination. It can be a critical ingredient ... but its usefulness in dependent upon the context in which it is used.

The major reason for considering computer applications (computers, workstations, local area networks, electronic mail) in schools is increased efficiency in information management and communications. Computer technology allows the administrator to find, process and disseminate information much more efficiently. When one administrator's workstation is tied to other workstations through a local area network that information management and communications process becomes even more efficient. Text and numeric data which once had to be assembled by hand can now be accessed using integrated word processing programs. Analysis of operating efficiency of the local school of school system can be done at the workstation using spreadsheet programs. Many repetitive tasks can be handled with minimum effort. Information from remote sites can be acquired easily through online database services. Higher quality printed and graphics materials can be produced. Properly managed, these resources allow the school administrator not only do what he/she did before; he/she can also do things which were not possible before. The school administrator can even change what the school does as well as how it is done. Implementing change turns out to be the major problem.

Change, Resistance, and Working Toward the Future

School administrators have begun to use technology in their operations but have not always considered the consequences for employees. In speaking about library administrator's move toward technology, Dakshinamorti (1985, p. 343) notes:

> Library organizations were in such a hurry to be included in this (technological) transformation that few contemplated the resulting effect of computer-aided systems on their employees.

Clearly, many school administrators have also fallen victim to the "get the equipment, then worry about the people" syndrome.

Reaction to change is the essential factor. The central issue is the management of that reaction to change. The new technologies are introducing new equipment, new processes and methods, new human relationships, new work assignments, or responsibilities. Almost all aspects of the school's work place may be changed. Yet, many people view change as unusual or terrible rather than as a regular part of life. Indeed, a great effort is put into making things stay the same. The pace of technological change is so rapid that many people feel that the technology is driving the work place, and that the equipment is taking precedence over the people.

People tend to view technology as either (1) bad—because it wastes resources, centralizes control, causes loss of personal freedom and dignity, and increases inequality, consumerism, deskilled jobs and eventual unemployment, or (2) good—because it causes increased personal freedom, more participation, more leisure time, more knowledge, and improved quality of life. In school settings examples of both views are found.

Technology does dramatically affect the way people work. Computers do jobs previously done by people (adding numbers, producing documents, creating graphics, etc.). Some individuals have developed skills in these areas which are now not only obsolete, but also more expensive than computer produced products. School staff may wonder if their jobs are going to be lost because the computer now does part of the work. Other staff may feel that they can not learn the new way to do work. Still other staff may feel devalued by the introduction of technology. Some staff may fear that their previously valued skills (such as manual filing, typing, organizing information, public contact) are going to be useless and, at the very least, unappreciated.

Figure 1 illustrates some potential responses to change as the individual evaluates change.

A change is proposed → *Individuals evaluate impact on themselves* → *Individuals respond*

Individuals can make the following evaluations and responses:

Evaluation	Response
Destructive	Resist opening
Threatening	Resist covertly
Uncertain	Tolerate
Positive	Support, participate

Andriole (1985, p 290)

Figure 1.

The evaluation of each individual is dependent on (1) the amount of information he/she has about the change, (2) the extent to which he/she can participate in the decision making process about the change, (3) How much the initiator of the change is trusted, and (4) the individual's past experience with change Thus, the amount of resistance to changes brought on by workstations and local area networks is largely dependent on the climate of the organization. Facing with the possibility of problems, obstacles, resistance, staff members develop their own denial systems. Denial also influences staff who will deny to themselves and others how they really feel about changes in operations, new equipment, different report forms, etc. Some steps can be taken to deal with the denial phenomena: encouraging discussion of the proposed changes, giving illustrations of how people might feel in meetings, and, above all, allowing time for free discussion. Find out how people really feel about their work situation and about the present equipment and technology they are using.

Administrators, teachers and support staff who have worked with computers in the school will be the most likely to accept the new systems and processes necessary to make school workstations and local area networks operate successfully. Even the most willing staff members who have very positive attitudes about computer related technology may be frustrated by the necessity of sitting at a terminal for long periods of time, waiting for the computer, or having something unexpected happen while they are working. People are concerned about costs — what will this change cost me in time and dollars? People want to know what the impact of the technology system will be on their work routines and their relationships with other workers (and their families). People may worry that they can not learn new ways.

Some school staff members have had experience with computers, on-line services, and word processing, and other staff (because of assignment or personal preference) have not. Unless the whole staff can be oriented and trained to some basic level of understanding, those who have the knowledge and experience will get more and the others will justifiably feel left out. This information-rich, information-poor dichotomy can work against well planned training efforts. If the staff is trained without regard to past experience, those who have experience may intimidate those who have no experience. Even the informal discussion which takes place prior to a staff development activity can have this intimidating effect when some staff members "share" their previous experiences with computers.

Since the introduction of technology into the school will affect working processes at all levels of the school, it is important that all staff be involved in the process of deciding on the uses of technology. The staff should be involved in the discussion of what areas of work the technology will impact, the variety of technological resources which could be utilized, and the process by which the choice of technologies is made. The process of involving the school staff in the planning of computer applications will depend on the planning style of the school management. Some organizations depend on detailed plans with highly structure schedules and carefully defined staff assignments. Other styles depend on a more information "brainstorming" method of planning. Training for all levels of staff about anticipated changes before they occur is essential — staff deal more effectively with known changes. Peters (1988, 637) issues the following warning:

> Implementation of the new integrated information technology-based systems is much more difficult than anyone dreamed. For one thing, it turns out that the installation of such systems is not primarily a matter of technology. It is a matter of organization. *Every* power relationship, inside and outside the firm, is affected by the installation of the new information technology systems. (Peter's emphasis)

Helping Empower Personnel Through the Change Process

School administrators do not have the freedom to become personal computer or network mavericks doing their own computing. For computer-based technology to work, all of the people with personal computers on their desks must be using the system to communicate and to improve the work process. No school can afford to support two information work process systems: the way we used to do it and the way we are supposed to do it now. The bulk of the administrator's time will be spent helping, encouraging, cajoling staff into moving more and more of their work operations into the computer.

Often the people who use typewriters, files, telephones have excellent insights as to the usefulness and limitations of these tools. Their ideas about improvements in the work flow and needed equipment capabilities can be helpful in determining what types of workstations and local area networks (if any) are needed. Allowing the staff to participate in planning of workstation and local area network systems, and in assisting with the selection of applications programs and standards, provides a way of enriching the work life of the staff. People can have power. People can acquire information from other individuals working at their peer level in other institutions; people can acquire new and more complex skills; people can make good decisions about what application to use for what purpose; finally, people can be rewarded.

For example, when a secretary has both a workstation with attached printer and a typewriter there is a tendency to still use the typewriter for envelops, memos and "short things." Yet applications software is available to allow envelop addressing, label making, and memo typing exactly like the typewriter. This software can be operated even while the author is working on the draft of this paper without changing the word processing program or the printer in any way. Even administrators are not free from the problem. Applications software now exists so that the secretary can maintain calendars for all of the staff, and create daily, or weekly calendar sheets which fit into the now popular Action Planner notebooks. In several years of consultation work, the author has yet to find a public school administrative group which took eagerly to the idea of maintaining a calendar in this way. People forget to tell the secretary about appointments, refused to allow the secretary to schedule appointments, ignored the daily printed calendars. If a common calendar system is used by a professional staff on a local area network, scheduling becomes much simpler and conflicts in meeting times are eliminated. The administrator can model appropriate behavior by instructing the secretary to keep his calendar, by referring to that calendar when making an appointment, and encouraging others to refer to his calendar on the network when they want to schedule events. OR he/she can continue to keep a desk calendar and not use the system.

Since staff will react to change, the school administrator can anticipate a number of potential problems and deal with them prior to any equipment choice or installation. Several steps are essential:

a) become knowledgeable about how computer-based systems work including the various utility programs: word processing, spreadsheets, databases
b) demonstrate knowledge of and concern about the work of all the staff
c) introduce the new system slowly, one step at a time
d) make sure everyone is using the same word processing, database, spreadsheet and telecommunications programs
e) make sure all your machines and systems can talk to each other ("connectivity")

a) Become Knowledgeable about How Computer-based Systems Work

The current and future administrator needs to have practical, school-oriented computer experience with word processing, spreadsheets and databases, as well as desktop publishing on various kinds of equipment in order to appreciate the potentials and the problems of these applications. Preservice or inservice training programs for school administrators should include:

Word Processing & Desktop Publishing Applications

School administrators communicate with a wide variety or publics through printed communication — letters, memorandums, public relations releases, etc. Much of this correspondence is repetitive as the school responds in the same way to similar situations. With capacities for form letters, merging of documents, and attractive layout of materials, word processing programs are ideally suited for such situations. Most current word processing programs allow for high quality text and graphic layouts and the creation of professional-level presentations (slides, overheads). Training programs should include experience with electronic mail systems which allow the administrator to distribute information within (and among) schools in electronic form.

Spreadsheets and Databases for Planning, Evaluation and Budgeting Applications

Spreadsheets are electronic ledger sheets. These programs help the library manager turn data into useful information for decision making. Planning, evaluation and budget analysis become much easier. Many of the more complex spreadsheets allow for the development of graphic presentations of the numerical data in a several graphic forms — bars, pie charts, etc. Such graphic representations are useful in reports and other presentation where masses of numbers simply confuse both the presenter and the audience. A powerful aspect of spreadsheet programs is the ability to create "what if" simulations allowing for detailed budget planning. Simple database programs allow the creation and updating of mailing lists, bibliographies, and inventories.

As Garland(1990) points out, this training process has become much easier as software has improved. It is easier to teach applications and have the participants succeed in their efforts in a relatively short time. The introduction of "icons" and pointing devices as well as more powerful graphic displays and printing have made very powerful, easy-to-use software packages available.

b) Demonstrate Knowledge of and Concern About the work of All Staff

The school administrator and his/her staff need to demonstrate that they know the jobs people do and care about the work the rest of the staff is doing. Often the question is: "Does the boss know and care about what I do?" Brod (1984,180) suggests several dimensions of present and future work design that need to be taken into account:

- knowledge: Are workers' skills being fully used in their present jobs? Is their work sufficiently challenging? Do they want more opportunity to learn new things?
- psychology: Are workers receiving sufficient recognition? Are need for achievement being met? Is there room for advancement?
- efficiency: Is the work exhausting? Is the degree of accuracy required excessive? Is sufficient information provided for maximum job effectiveness?
- task structure: Is there sufficient variety in the work? Are workers given enough scope to use their own initiative? Are they provided with a chance to make planning suggestions?
- ethics: Does management look after the interests of its employees? Do managers convey respect for the skills of the workers? How do workers perceive top management and its goals?

The larger the organization, the more important it is to make public notice of the work of all subordinates on the staff and the contribution that work makes to the overall goals and objectives of the school. That is the nature of management—managers not do work, they get others to work. As Hutchins (1947, 138) said long ago:

> The administrator should never do anything he does not have to do, because the things he will have to do are so numerous that he can not possibly have time to do them ... He should never do anything he can get anybody to do for him. He should have the largest number of good associates he can find.

This empowering of others is essential if the power of technology is to be directed toward good ends. As the planning process for technological change proceeds everyone should be able to identify what their role will be in the emerging structure. When everyone's role and responsibilities are clear, the staff can function more effectively. Part of the staff development process should include a review of the school's history of change as a means of acknowledging how this staff (and their predecessors) survived and perhaps even thrived on change. The administrator needs to be frank about his/her motives in suggesting the change; what are the "rewards" or benefits which this change will bring about and why are they important to the school?

A part of this empowerment process is the development and announcement of a consistent, humane policy on how the technology of workstation and local area networks is going to be used. This policy needs to clearly state that the school recognizes the psychological needs of the staff and takes those factors into account in designing work and assigning that work. Jobs are to match people, not people jobs.

People vary in their ability to process information with any technology (pencil, typewriter, telephone, computer). The employee and his/her immediate supervisor are the best judges of the capabilities in this area. Everyone will not produce the same amount of data (or completed telephone calls). People also vary in their ability to "stick" with a task. Some individuals are not able to monitor their own work pace and need assistance in knowing when to stop and take a break. Some individuals will work a high speed in short bursts...others will be slower but will continue a consistent rate of production over a longer period of time. Institution policy should allow for such variances.

The basic principle of policy should be a balance between productivity and worker well-being. The worker's well-being can be enhanced on the job design allows for some type of challenge to be built into workstation and local area network tasks. Work that is too simple breeds boredom and may even cause hostility. Where workers have the opportunity to learn new skills and assume new responsibilities; they have a sense of contributing to the organization as a whole; they are empowered.

c) Introduce the New Systems Selectively, Slowly, and One Step at a Time

One major reassurance which can be repeatedly stressed is that learning the new systems will take some time and that work flow may be slowed during that process. All too often a technology is introduced to "improve work flow performance" which it may do in the longer run, but can not do until people know how to use it and are comfortable with it. Since schools must remain open specific hours, and continue to serve their clients, school administrators have several options during the training period:

- additional temporary staff to handle some functions while the staff is in training
- discovering staff members who are very proficient at using the system, and temporarily reassigning them as trainers.
- hiring outside organizations to do training for staff members which offers the incentive of academic credit (community college or university) or ease of scheduling (after work).

Since not everyone will enthusiastically support change, the manager can use the planning period prior to installing a computer workstation or local area network system to identify those work groups in the school who are either most interested or most in need of the services such a system can provide. Few organizations will be able to bring their system up "all at once." If the school administrator and staff do a careful analysis of work groups and needs, they will be able to identify areas where current information flow is slowed (or even blocked) as well as areas where the departmental or work group is eager to get involved. Starting with areas of critical need or eagerness, the school director can involve several units in detailed planning, pilot operations, training seminars, etc.

When departments or work groups do a good job utilizing the workstations and the local area network, the school manager needs to be sure that everyone (staff, board of trustees, etc.) knows about how well they are doing. This "island culture" idea has been successfully used in many institutions and is celebrated in Peters and Austin's A Passion for Excellence. (1985). If the people who are using the system seem to be able to make it work for them in their jobs and seem to be enjoying it, other staff groups may want to get into the act. Many agencies utilize an "electronic newsletter" on the local area network to announce new applications, to praise effective uses, and to share solutions to problems bothering everyone on the system.

d) Establish a Policy that Everyone Use the Same Word Processing, Database, Spreadsheet and Telecommunications Programs on the Same Kind of Equipment

Many times schools acquire specific computer hardware and software to do specific jobs. Personal computers are not always of the same type and software is often not designed to be "shared" among different machines. For example, in one school the author knows, computing started as a "public access" function when most of the area schools had Apple computers and software. The logical configuration was a group of Apple II computers and related applications software including Appleworks, a word processor, Visicalc, and some "learning" programs and games. Later the administrative office acquired a PC type machine for word processing and eventually spreadsheet applications. As administrative word processing increased departments acquired various models of PC clones, word processing packages available as "shareware" from ads in computing magazines. Now, interest in desktop publishing has caused the school system to acquire a Macintosh with a laser printer and software for designing page layouts. Within a three year period, the library has acquired 3 types of computers and about six types of word processing software.

Planning prior to acquiring computer technology allows the administrator to establish policies on equipment and software acquisition. The point of such a policy is to insure that information can be communicated in the easiest possible way. Where the equipment and software is compatible, the process of sharing form letters, spreadsheet templates, and electronic mail becomes much simpler. As the school develops the need for local area networks (LANs), connecting machines of the same type is cheaper and easier.

Sometimes administrators will ask, "what do we do with the old equipment and programs?" Very often the answer has to be, "discard them." If equipment and software is retained too long, there will be a growing incompatibility between old and new systems. Much of the problem about computer equipment comes from the fact it is considered to be "fixed assets." In fact, the pace of change and the declining cost of equipment mean that computers should be thought of as "consumables." When the first system is purchased, plans should already be under way for the next system.

The administrator must aware of what Hannigan (1988, 216) calls "the Curve of Nonsatisfaction" as computer technology continues to be used. This move toward dissatisfaction with the system(s) is usually made up of:

- increased breakdown of hardware
- increased software failures
- increased data loss
- voiced dissatisfaction with software
- reported system slowdowns
- requests for additional and more complex data manipulation and reporting
- suggestions for purchase of peripherals
- increased requests for machine replacement
- requests for newer or enhanced version of software
- reports of frustration because the system does not meet user needs
- breakthrough technology making current architecture obsolete

Since this curve of dissatisfaction seems inevitable, the administrator must evaluate system operations from the users' perspective and try to anticipate staff concerns and budget requirements for replacement, enhancement, and conversion to newer systems. Information on how the system is working and what particular problems are occurring can provide feedback to long range planning for solutions to problems before they get out of hand.

e) Make Sure All Your Machines and Systems Can Talk to Each Other ("connectivity")

Current educational uses of telecommunications and electronic mail depend on combining computer and telephone communication. In business and industry, communication of information within facilities and in regions is now handled by local area networks. Such networks allow massive amounts of information to be stored on a central file server which is accessible to everyone whose machine in connected to that network. The key word is connected. If the administrator has standardized equipment and software throughout his school (or system), local area network connection becomes much easier. Usually such connections require only the installation of cards, cables and programs. If the administrator faces a variety of equipment and software, some standardization is going to be required. In addition, there will be additional expense involved in creating connections between various types of machines.

In the future most information will be created, stored, and transmitted in digital form (including those bulky student files). As the administrator leads his staff and faculty to use computer technology, he/she must keep asking the question, "Can our systems communicate within the school, across the system, and in the region?"

Conclusion

Often managers see computer technology as a means of cost reduction and higher quality of work. Properly trained and motivated staff members want to, and will provide improved work. Hopefully school administrators planning technological change in schools will involve themselves in using the technologies, bring to their jobs some understanding and empathy for the individuals doing the work, and have some understanding of the need for policy about equipment, software, and communication between systems. Being supervised by a person who is knowledgeable about the technology and with whom you can communicate can make all of the difference.

References

Andriole, S. J. (1985). The future of information processing technology. Princeton, NJ: Petrocelli Books.

Bland, S. 1987. The Media Lab. New York: Viking.

Brod, C. 1984. Technostress: the human cost of the computer revolution. Menlo Park, CA: Addison-Wesley.

Dakshinamurti, G. B. 1985. Automation's effect on library personnel. Canadian Library Journal, December, 343-351.

Garland, V. E. 1990. Software trends in the training of school administrators. T.H.E. Journal. 18:5, November, 86-88.

Hannigan, J. A. 1988. An expanded managerial role in a microcomputer environment. In Intner, S.S. and Hannigan, J. A. The library microcomputer environment: management issues. Phoenix, AZ: Oryx Press.

Hutchins, R. M. 1947. The administrator. In Works of the mind. edited by R. B. Heywood. Chicago: University of Chicago Press, 135-156.

Kinnaman, D. E. 1990. The next decade: what the future holds. Technology & Learning, 11:1, September, 42-49.

Peters, T. J. 1988. Thriving on chaos: Handbook for a management revolution. Perennial Library Edition. New York: Harper & Row.

Peters, T. J. and N. Austin. 1985. A passion for excellence; The leadership difference. New York: Warner Books.

Seiler, L. H. 1989. The Future of the Scholarly Journal. Academic Computing. 4:1, September, 14-16, 66-69.

The workstation revolution. Scientific American, 260:5, w1-w9.

Special Presentation

Earth Lab: Application of LAN Technology to School Restructuring

Denis Newman, Bolt Beranek and Newman Inc.
Paul Reese, Ralph Bunche School
New York City

Earth Lab is a unique LAN-based school environment for elementary and middle schools designed for communication, sharing data, and collaborative learning across the curriculum. This year the technology is being used to support a new school-within-the-school in which a group of teachers collaboratively teach grades 3-6. Earth Lab brings together Apple IIs and Macintosh computers throughout the mini-school on an AppleShare fileserver. The technology supports collaboration among teachers as well as among the students. We will report on this unusual experiment in technology support for restructuring. Particularly relevant is the functionality for teachers who are: developing curricula involving teacher collaboration across disciplinary boundaries; setting up fileserver workspaces for students or work groups which are available across grade levels; sending messages to individuals or groups of students; communicating with colleagues in the school and elsewhere in the world.

Special Presentation—ISTE

Hypermedia and Multimedia—Current Trends in Education

Special Presentation
MediaText: The Brownie Instamatic of Multimedia

Elliot Soloway
University of Michigan

Abstract not available at press time.

Special Presentation
HyperCard/HyperStudio Based Curriculum for Bilingual Instruction

Michael Reed
Peach Springs Elementary School
Peach Springs, Arizona

Abstract not available at press time.

Projects
Algebra, Calculus, and Statistics

Math for the 90s: Teletraining Project

Catherine F. Conway
Boston Public Schools, Massachusetts

Abstract

Math for the 90's is a two year professional development program designed to introduce Boston Public School teachers (K-12) to the mathematics standards published by the National Council of the Teachers of Mathematics in 1989. Funded by the U.S. Department of Education (Dwight D. Eisenhower—Title II Training Grant), this project is a collaboration between the Boston Public School's Departments of Mathematics and the Technology Development.

Using a "distance learning" model, teachers receive "hands on" math instruction from outstanding math educators and teacher-leaders from their school. Each participant receives a small stipend and manipulatives to use with his/her students.

This program has been enthusiastically received by students,teachers and administrators. In many schools, Math for the 90's served as a catalyst, propelling faculty members into grant-writing and further professional development opportunities.

Using "distance learning", elementary and middle school teachers received four hours of math instruction from Dr. Robert Willcutt of Boston University via cable TV. Participants shared ideas with the on-screen instructor and with each other, using a telephone. Responses and comments were part of the program itself. Each on-screen session was followed by an additional hour of instruction provided by a teacher-leader from that building who was a member of the EMAT program (Elementary Math and Technology—funded by the National Science Foundation).

The content at the elementary level focused on student learning styles and on math problem solving techniques using Cuisenaire rods and the Powers of Ten blocks. Throughout the "live" training sessions, participants were able to see these lessons "in action" in Boston classrooms through the magic of videotape.

At the middle school level, teachers learned more about the adolescent as a learner, and how to teach geometry, fractions, decimals and pre-algebra concepts using: Cuisenaire rods, the Fraction Factory and Algebra Tiles.

In the fall of 1990, 350 elementary teachers, who participated in the Math for the 90's, took part in a fifteen hour math professional development program (Math for the 90's Part II) held at their own schools, using Massachusetts State Department of Education funds. These sessions were taught by the teacher-leaders who had facilitated their previous instruction. In addition to inservice credit, participants received a collection of math lessons (developed by teachers from the EMAT Project) which incorporate the use of manipulatives and computers in daily math instruction.

TapeMeasure: Video as a Source of Data in Student-Designed Investigations

Andee Rubin
TERC
Cambridge, Massachusetts

Abstract

We have designed, implemented and field-tested a system that allows students to capture and analyze data from videotape. They are provided with three on-screen measurement tools for measuring length, angle, and time, as well as two "mark-up" tools to draw dots or lines as references for later measurements. The measurement tools generate numbers which students can easily enter into their own datasheet; a classroom's worth of datasheets can be merged into a standard spreadsheet so that students can make graphs using any graphing program.

Our field test consisted of a two-month long project in a seventh-and-eighth grade classroom in an urban school district. The teacher presented the students with the problem of investigating "What variables affect how fast you run?" Students carried out a formal experiment by choosing a small set of variables, determining the details of running and taping that would give them best access to those variables, making measurements both on the videotapes and in class, and analyzing associations among the variables. Their final conclusions were : 1) stride length is most closely related to running speed and 2) a more accurate model of the experiment would require incorporating more than two variables at a time.

Algebra Processors: An Intellectual Companion for Calculus Students

Peter Flanagan-Hyde
Phoenix Country Day School, Arizona

Computers and calculators have long permitted students to perform numerical calculations very quickly. Not until recently, however, have inexpensive computers been able to process and manipulate symbolic expressions of the kind which are typical in secondary school mathematics. The computer can become more of an intellectual companion for the student by permitting the student to focus on the larger picture and not get distracted by the mechanics of the algebra involved. Students are encouraged to take a more inductive, "what if…" approach to the work, given the ability to rapidly calculate, draw graphs, or animate graphs. Students are encouraged to represent the same idea in different formats: as an equation, as a graph, or as an infinite series.

Student work will be presented in two general areas:

- Basic capabilities of an algebra processor. Simplifying expressions, solving equations, expressions with trig, log, and exponential functions, finding derivatives, finding integrals, calculating definite integrals, and evaluating Taylor series, among others.
- Interactive graphics capabilities of the algebra processor. Graphing a function, graphing the derivative of a function, demonstrating tangent lines, graphing in polar coordinates, graphing of parametrically defined functions, graphing of piecewise functions, and the animation of graphs.

Special Presentation
Computer Science Curriculum Using the Logo Language

Ron Place, Chuck Auck, Mike Sansing
Logo Curriculum Publishers
Colorado Springs, Colorado

One of the issues today is which language is best for introductory level programming courses. Over the past five years, Chuck Auck, Ron Place and Mike Sansing of Colorado Springs, Colorado have developed a full semester computer science curriculum using Logo as the language of choice. This 423 page curriculum emphasizes problem solving strategies, critical thinking skills and solid principles of computer science at the secondary level.

This curriculum is a turnkey approach providing all the necessary materials for an immediate and successful implementation. The graphics environment generates enthusiasm. The students are truly excited about attending class and begin working on their assignments before the bell rings. This curriculum has been taught to over six hundred students. These students progressed through the higher level programming courses at a much faster pace with great success.

Paper

Beyond Computer Literacy for School Administrators

Charles F. Adams
Wichita State University

Abstract

An evolution from offering courses where the computer itself is the subject of instruction to those where the computer becomes an instrumental means of teaching traditional course content is advocated in this paper. An example is provided for restructuring a required graduate course in educational administration which uses the computer as the central way of teaching educational resource planning. (Keywords: administrator training, computer literacy, administrator competencies)

For those employed in graduate programs that prepare school administrators, the questions of whether, what, how and when to teach management uses of the computer have persisted for the past two decades. Recently, a description of university programs in educational administration across the United States was presented by Spuck and Bozeman (1988). In their survey they found that approximately 25% of the universities that award doctoral degrees required a course in computers for either the masters, doctoral degree or certification for the principalship; and that an additional 30-50% would accept such a course as an optional elective (Spuck and Bozeman, p3). Although many reasons were offered for the relatively low priority given teaching administrative uses of the computer, the major one was that college professors themselves were not frequent computer users.

The Wichita State University could be described as typical of the "average" institution depicted in the aforementioned study, although it does not yet offer the doctoral degree. Only one, and in recent years two, of the professors had sufficient computer knowledge to teach a course in the graduate program for school administrators. This introductory - literacy type course, although an optional course, has been strongly recommended for most students. The content and objectives of this course have evolved since its inception five years ago from introductory computer literacy to skill development in using current popular software packages. Part of the continuing changes in the course could be attributed to the changes that have occurred in the hardware and software industries. The upgrading of microcomputers from the first generation Apples and Radio Shacks to current 80386 and 68020 processor machines and from simple word processors, file managers and spreadsheets to integrated databases, report generators, desktop publishing software, etc. has occurred along with student exposure to telecommunications and specialized school accounting and student information management software products. Also, a much larger percentage of students enrolling in the program have had limited experience in using microcomputers, negating the need to focus on many of the fundamental literacy skills that were the staple features of the course in it first few years.

At the present time, the Department of Educational Administration and Supervision is planning to eliminate this course in computer literacy for school administrators. This action may seem contradictory to the widely held belief that administrators need to be well equipped to manage schools in the information age. All new entrants into the profession should be knowledgeable information managers including having the ability to use the computer to increase their and their staff's productivity (Bosch, 1988). The WSU educational administration faculty concur with this proposition but believe there is a more appropriate way to assure that prospective administrators acquire the essential computer knowledge and skills needed to manage schools.

The educational administration staff has just completed planning for a new doctoral program in educational administration. As part of this planning, the existing master's and specialists degree programs were also revised. The conceptual base for all program and curricular revisions was administrative competencies. During the planning period professors in the department deliberated at length as to the knowledge and skills in computer based management information that would be required by present and future practicing administrators. The following competency areas were seen as essential priorities for inclusion in the preparation programs:

1. ability to design school data structures and use database software for a wide variety of school information needs
2. ability to design and produce decision relevant reports from the school database, both routine and exception/specialty reports
3. knowledge of sources of information in public databases and the skill to access them
4. plan and implement instructional management systems
5. have functional skill in using generic software packages to support decision making (e.g., spreadsheets, word processors, desk-top publishing, project planning, etc.)
6. establish data security procedures
7. ability to plan the acquisition of appropriate, cost-effective computer hardware, software and telecommunications systems

These competencies will be integral to the core courses in the new preparation programs. Rather than have a separate course specifically designed for teaching a set of competencies about computers, it was decided that those competencies could be learned by students at the same time they were learning the fundamental tasks of administration such as budget building and accounting, developing teaching and staff work schedules and program evaluation. Thus the instructional strategy to be followed is one of integrating computer use in the delivery of instruction and required learning activities within the core administration courses.

One new course developed at the master's degree level and required for principal certification is Facilities and Resource Management. It typifies the intent to teach fundamental administration functions by using computer based activities. A combination of lecture/seminar activities and computer lab activities are planned for this course which will be offered for the first time in the Fall of 1991. A brief overview of the course content and teaching-learning activities is presented as an example of integrating computer instruction with the traditional administration subject matter.

Content Outline	*Instructional Method*
I. Manage Essential School Information	
A. Identify essential data categories for the school	Using computer database software, design integrated file structures to account for 1) student biographical, achievement, attendance, and other pertinent data; 2) fixed assets of the building.
B. Develop and maintain accurate record-keeping systems for pertinent school data	Using computer software, generate recurring and exception reports from the school databases.
C. Develop detailed tactical plans	Using computer software, develop a project/task timeline, resource allocation and monitoring plan for a specific project.
II. Building or Unit Resource Allocation	
A. Establish a rational planning system to determine staffing, supplies, equipment, and plant operation needs and translate into budget allocations	
B. Plan a detailed annual budget for the building/unit	Using an appropriate accounting software package or spreadsheet software, develop a function/object line-item budget.
C. Account for budget expenditures	Using an accounting software package, monitor purchase orders, expenditures, and budget code.
D. Understand Kansas bid laws and conventional purchasing practices balances.	
E. Using enrollment and program data, develop a staff allocation and course scheduling plan	Using an appropriate software package, develop class, room, teacher and student schedules.

Content Outline	*Instructional Method*
III. Facilities Management	
A. Define building needs in terms of students and programs	Using spreadsheet software, develop a 5 year enrollment forecast.
B. Define and monitor work tasks for custodial staff and plan summer maintenance schedules	(See I.C. above)
C. Develop maintenance procedures for grounds, playgrounds, outside teaching stations, etc.	
D. Plan and monitor a school food service program	(See II.C. & D. above)
E. Plan and schedule a program of extra-curricular student activities	
1. appropriate programs for student population	(SEE II.E. above)
2. procedures for staff supervision of activities	(SEE II. A. above)
3. develop a budgeting and accounting system for student activity program monies	(SEE II.C. & D. above)

The use of simulated problem situations from school operations forms the core of teaching-learning activities in this course. Using the computer as a tool to solve these simulated problems will not only teach the essential computer related skills but also demonstrate its versatility and productivity enhancing capability to new educational administrators.

In a similar vein, planning efforts are under-way to incorporate computer skills in other educational administration courses such as personnel administration, school plant planning and curriculum development. Admittedly, these revisions in the curriculum are only in the planning stages and results are yet to be determined. Careful monitoring of implementation and outcomes of the new curricular approaches is necessary. To assist in evaluating the effectiveness of the revised preparation program, a database will be established to record student achievement in various competency areas. The entire set of required competencies for each program level (master's, specialist's and doctoral) will be one major file in the database. At the completion of each course, students will be evaluated in terms of the degree to which each competency has been acquired. Records of student competency assessment will be the second major file in the database. Competencies in using the computer will be one sub-set of the competencies monitored and recorded. Thus information on student achievement will be available to assess the effectiveness of instructional programs as well as generating a competency profile on each graduate.

It is almost axiomatic today, that educational administrators must be knowledgeable about and skilled in using the computer. However, requiring prospective administrators to take a basic course in using the microcomputer leaves the student with the problem of applying the skills until they are in an administrative position. Teaching a wide variety of administrative competencies via the computer demonstrates their applicability to actual administrative situations (through simulation). By incorporating using computers in a variety of more traditional educational administration courses, the faculty at WSU believes that their graduates will be able to employ computers in appropriate problem solving ways in all areas of school management.

References

Bosch, K.A. (1988) Microcomputer literacy training for school administrators. *Journal of Research on Computing in education*. 20(4) p. 331-338.

Spuck, D.W. and Bozeman, W.C. (1988) Training school administrators in computer use. *Journal of Research on Computing in Education*. 21(2) p. 229-239.

Dr. Charles F. Adams is an Assistant Professor of Educational Administration at The Wichita State University where he teaches strategic planning and management information systems for school administration. He is also the Vice President of the Institute for Future Systems Research. (Address: Campus Box 123, Wichita State University, Wichita, KS 67208)

Paper

Competencies Necessary For An Integrated Microcomputer Application Course

Dr. Joseph C. Otto
California State University, Los Angeles

Abstract

The research for this study was conducted to develop a list of competencies that should make up the course content of a single microcomputer applications course dealing with word processing, spreadsheet applications, and database management. A group of experts in the field of computer applications software reached a consensus, via the Delphi technique, concerning what should be included in the content of an applications course.

Computers are used in all types of environments. They are used in businesses, governmental agencies, educational institutions, and in the home. Computers are used to provide an incredible range of services to everyone in our society.

Computers are used for all types of circumstances and in all types of situations. Consequently, the demand for computer-knowledgeable people is always increasing.

Since most people will interact with computers at their place of employment, a knowledge of computers and business application software is becoming extremely important in both job performance and career development.

Many people believe that the true impact of the computer is yet to be felt. Computers get cheaper and yet more powerful every year, which means that each year a better product is available for less money. Consequently, computers are being applied in new areas, and the demand for computer-knowledgeable people is always increasing. The ever expanding use of computers, constant changes in office technology, advanced systems and procedures, the changing nature of office jobs, growing opportunities for career advancement, and new careers related to all types of information processing have brought about the need for additional basic skills and knowledge for employment in today's automated office. This poses a problem for universities, colleges, and other institutions that instruct in the computer area. If the knowledge of computers is that important, the institutions of this country have a significant role to play in computer education.

The need for computer software applications education for all students is evident. The question that does exist, however, deals with the content of a computer class on today's campuses and which competencies of word processing, spreadsheets/graphics, and database management should be included in these application courses. It is imperative that students learn to effectively use software in the following major areas: word processing, spreadsheet, and database software.

Background of the Study

With the increased use of computers and microcomputers, the learning of computer literacy and computer applications is becoming more important. Most of the research conducted has been in the area of competency in word processing software or in computer literacy. Research in word processing has been in the area of overall curriculum rather than in the components to be included in a course. Research in the areas of spreadsheet and database application software is limited.

Problem

The problem that this study addressed was to identify, by utilizing the Delphi technique, the competencies necessary in the course content of an integrated microcomputer applications course in a post-secondary institution.

Procedures

To address this problem, a panel of experts evaluated a list of competencies related to an integrated applications course and rated the importance of each competency. For the purposes of this study, the panel of experts is defined as a group of individuals chosen on the following criteria: has been published in the computer applications area OR completed a doctoral dissertation dealing with computer applications OR has published a computer software applications textbook. The panel of experts in microcomputer applications rated the relative importance of 285 competencies in order to identify those competencies necessary in a microcomputer applications course. In three Delphi iterations they generated the final list of 103 competencies essential for an applications course, and 181 competencies determined to be of some importance for a computer applications course.

Findings

Table 1 shows the competencies essential for inclusion in the word processing portion of a single microcomputer applications course involving word processing, spreadsheets/graphics, and database management.

Table 1
Essential Word Processing Competencies

Competency
Keyboarding
Creating a document
Function keys
Cursor movement and scrolling keys
Entering text
Using wraparound to enter text
Using the return or enter key
Keyboarding capital letters
Editing
Correcting errors by backspacing
Deleting left
Deleting right
Inserting and deleting characters
Inserting and deleting words
Inserting and deleting sentences
Deleting to end of line
Deleting to end of page
Deleting a block of text
Reformatting text
Moving text
Moving a block of text
Copying text
Copying a block of text
Format
Indenting paragraphs
Setting left and right margins
Setting top and bottom margins
Inserting blank lines
File Management
Storing document files to disk
Storing document files to disk and quitting
Retrieving document files from storage
Deleting document files from storage
Renaming document files
Displaying document files on disk storage
Loading or saving part of a document file
Copying a document file
Printing
Print a document
Miscellaneous
Help menu

Table 2 shows the competencies essential for inclusion in the spreadsheet section of a microcomputer applications course.

Table 2
Essential Spreadsheet Competencies

Competency
Keyboarding
Spreadsheet status line and menu line
Function keys
Cursor movement keys
Entering labels and values
Entering data
Entering formulas to perform addition, subtraction, multiplication, and division
Entering formulas to perform combined computations
Editing
Changing cell contents by replacement of single cell entries
Erasing (blanking) cell contents
Deleting row(s) or column(s)
Inserting row(s) or column(s)
Correcting by replacement
Revising by deleting characters
Revising by inserting characters
Correcting errors in slash commands
File Management
Saving a spreadsheet file on disk
Storing document file to disk and quitting
Retrieving a spreadsheet file from disk storage
Deleting a spreadsheet file from disk storage
Format
Aligning text on the left, right, or centering text
Setting decimal places for a cell
Setting decimal places for a range
Setting decimal places for the entire spreadsheet
Displaying integer values
Replication
Copying a cell
Copying a range of cells
Copying text from a single cell into multiple cells
Copying a formula into a cell
Functional Formulas
Computing sums and counts
Computing averages
Printing
Printing the spreadsheet
Printing the spreadsheet cell contents
Printing a section of the spreadsheet
Miscellaneous
Help menu

Table 3 shows the competencies essential for inclusion in the database management portion of a single microcomputer applications course.

Table 3
Essential Database Competencies

Competency

Keyboarding
- Function keys
- Cursor movement keys
- Entering commands
- Entering records into a database
- Clearing the screen
- Creating a database file
- Creating the database file structure
- Defining the field names
- Defining the field types
- Defining the width and decimal positions

Editing
- Selecting a record
- Adding records to a file
- Inserting a record in a file
- Deleting records from a file
- Changing records in a file
- Modifying field contents
- Modifying record contents
- Deleting fields from the directory

Displaying
- Displaying selected records
- Displaying all fields
- Displaying selected fields
- Displaying a certain record
- Displaying using relational operators

Sorting
- Sorting records in descending or ascending sequence
- Sorting on more than one field

Printing
- Creating a report
- Designing the output
- Printing the report

File Management
- Displaying database files directory on disk storage
- Retrieving database files from diskette
- Deleting database files from diskette
- Making backup files of database files
- Copying a database file

Miscellaneous
- Help menu

Table 4 shows the competencies determined to be of some importance for inclusion in the word processing portion of a single microcomputer applications course.

Table 4
Important Word Processing Competencies

Competency

Keyboarding
- Status Line, Ruler Line, and Menu Line
- Goto Command
- Hard Spaces and Hard Returns
- Underlining Text
- Centering Headings
- Centering Lines
- Boldfacing
- Numbered Lists
- Creating Forms

Editing
- Undeleting Text
- Keyboarding Over Existing Text
- Searching for Words
- Searching and Replacing Text With or Without Confirm
- Backward Search
- Searching with Upper/Lower Case Capability
- Searching Using Wild Characters
- Searching Using Whole Word Capability
- Working with Blocks of Text
- Writing and Reading a Block of Text

Format
- Margin Release
- Justifying Margins
- Adjusting Margins
- Setting Lines/Inch
- Setting Characters/Inch
- Setting Tabs
- Tab Align
- Working With Indent Tabs
- Indenting Left and Right
- Setting Decimal Tabs
- Working With Decimal Tabs
- Adjusting Word Wrap
- Double Spacing
- Adjusting Line Spacing
- Working With Line Draw
- Centering Headings Over Columns
- Inserting Page Breaks
- Super/Subscript
- Centering Page Top to Bottom

Table 4
Important Word Processing Competencies
(continued)

File Management
- Storing Document Files to Disk With Overwrite
- Storing Document Files to Disk and Resuming
- Storing Document Files to Disk and Switching to Another File
- Changing Data Drive
- Abandoning a Document File
- Consolidating Two Document Files
- Locking or Protecting a File

Printing
- Printing Part of a Document
- Printing Current Page
- Print Screen
- Printing to Disk
- Printer Setup
- Conditional End of Page
- Printing More Than One Document
- Working With Different Printer Fonts and Printer Enhancements
- Printing With Hand Fed-Paper or Sheet Feeder
- Working With Printer Commands
- Working With Automatic Pagination
- Page Number Position Set-up
- Printing More Than One Copy
- Printing From Disk
- Printing Envelopes
- Printing Different Sized Paper

Miscellaneous
- Performing Spelling Functions
- Performing Thesaurus Functions
- Performing Grammar Functions
- Performing Style Functions
- Performing Math Functions
- Working With Columns and Column Functions
- Inserting Text Variables
- Boilerplating
- Composing Text
- Cancel Key
- Upper/Lower Case Convert
- Defining Macros
- Working With Text or Data Files
- Working With Windows
- Working With Orphans
- Working With More Than One Document File on the Screen
- Endnotes or Footnotes
- Merging Two or More Document Files
- Headers and Footers
- Working With Hyphenation
- Setting up Automatic Indexing
- Setting up Automatic Table of Contents
- Sorting Function

Table 5 shows the competencies determined to be of some importance for inclusion in the spreadsheet portion of a single microcomputer applications course.

Table 5
Important Spreadsheet Competencies

Competency

Keyboarding
- GoTo Command
- Entering Values with Decimal Default
- Formulas with Relative and/or Absolute References
- Using the Pointer to Enter Formulas

Editing
- Erasing (Blanking) a Rectangular Range
- Clearing the Entire Spreadsheet from Memory
- Revising by Striking over Characters
- Moving a Row
- Moving a Column
- Moving a Range

File Management
- Changing Data Drive
- Saving a Spreadsheet File on Disk with Overwrite
- Storing Document File to Disk and Resuming
- Storing Document File to Disk and Switching to Another File
- Displaying Spreadsheet Files on Disk Storage
- Consolidating Two Spreadsheet Files
- Loading or Saving Part of a Spreadsheet File
- Copying a Spreadsheet File
- Renaming a Spreadsheet File

Format
- Global Format
- Column or Row Format
- Adjusting Column Width
- Including the Dollar Sign with the Value
- Including Commas with the Value
- Designing Spreadsheet Layout
- User-Defined Format
- Column Total Lines and Underscores

Replication
- Copying a Formula into a Range
- Copying a Label into a Range of Cells
- Copying a Formula without Adjustment
- Filling In a Data Table

Functional Formulas
- Computing Counts with Blank Cells
- Finding the Largest Value
- Finding the Smallest Value
- Logical Functions

Table 5
Important Spreadsheet Competencies
(continued)

- Mathematical and Engineering/Scientific Functions
- Statistical and Financial Functions
- Calendar Functions
- Using IF Functions to Determine Status of a Value
- Using IF Functions to Make Computation(s)

Printing
- Printing a Report Without the Spreadsheet Borders
- Print Screen
- Printing to Disk
- Printer Setup

Miscellaneous
- Horizontal and/or Vertical Titles
- Horizontal or Vertical Windows
- Synchronized or Unsynchronized Split-Wise Scroll
- Building a Master Spreadsheet without Formulas
- Building a Master Spreadsheet with Formulas
- Protected/Unprotected Cell Entries
- Manual or Automatic Recalculation
- Defining Macros
- Sorting Function

Graphics
- Using Spreadsheet Data to Construct a Graph
- Specifying Data to be Graphed
- Defining Variables
- Formatting the Graph
- Viewing the Graph on the Screen
- Choosing Type of Graph
- Headings: Main, Sub, and Column
- Naming a Graph
- Saving a Graph
- Printing a Graph
- Creating Pie Charts
- Creating Line Graphs
- Creating Bar Graphs
- Creating Stacked Bar Graphs
- Creating Area Fill Graphs
- Creating XY Graphs
- Adding Labels to a Graph
- Changing the Graph
- Changing Graph Formats
- Creating Multiline/Multibar Graphs

Table 6 shows the competencies determined to be of some importance for inclusion in the database management portion of a single microcomputer applications course.

Table 6
Important Database Competencies

Competency

Editing
- Packing a File
- Browsing Through a File
- Combining Fields and Records from Two Database Files
- Correcting Commands
- Modifying the Database File Structure
- Updating a File With Data From Another File

Displaying
- Displaying All of the Records
- Displaying Next Record
- Displaying Using Logical Operators
- Displaying a Count of Records
- Displaying a Sum of a Field
- Displaying an Average of a Field
- Displaying a Sorted File

Sorting
- Sorting Selected Records From a File

Printing
- Preparing a Report Using Selected Records From a File
- Printing Reports With Subtotals
- Printing in Report Form
- Setting Left and Right Margins
- Setting Top and Bottom Margins

File Management
- Renaming Database File
- Changing Default Drive
- Protecting a Database File
- Entering and Running Command Files
- Editing and Correcting Command Files

Recommendations

Following a review of the findings of this study, the following recommendations can be made:

1. Software developers must strive to include all of the necessary microcomputer application competencies in their software and the materials that accompany their products.
2. Schools that are anticipating the addition of a course in microcomputer applications must be sure to include the essential competencies, as identified by this study, in this type of course.
3. Curriculum planners who are evaluating current course offerings, or content of courses currently offered, must verify that the essential competencies for a microcomputer applications course, as determined by the experts, be included in their curriculum.
4. Textbook authors should be sure to include the competencies identified by this study in the products they develop for a microcomputer applications course.

Projects
Computer Education Leadership and Graphic Art

Animation and Interactive Video in Multi Media Health Science Instruction

James Wall, John Strange
Department of Physical Therapy and Department of Education
University of South Alabama

Abstract

Interactive media offers enormous potential for the development of teaching materials for students in the health sciences. The major drawback with this technology has been the lack of appropriate programs, primarily because of the lack of programming expertise on the part of the faculty. Recently authoring systems which are extremely powerful and yet still relatively simple to use have been produced. As an example of the use of this technology, a number of interactive programs have been developed for use in the Physical Therapy curriculum. The computer screen of an Apple Macintosh is used to display textual material, graphics and animation while a second screen displays the output from a videodisk player. All the materials are interlinked and under the control of the student by way of the computer. A gait tutorial uses flip page type animations to show, for example, the spatial parameters, while more complex animations are used for showing such elements as the phasic activity of the muscles and the ground reaction forces. A videodisk is used to show actual walking sequences. A second program, being developed jointly with the University of Tennessee, Memphis, is designed to assist students review material on abnormal reflex development.

Guidelines for Better Screen Design

Penelope Semrau
California State University
Los Angeles, CA

Abstract

Many teachers today are creating their own educational software with authoring tools like *HyperCard*. Even though one has mastered the mechanics of *HyperCard*, making the software motivating is another matter. The integration of visuals in educational software creates a greater impact on the learner than the use of words alone. What is it that makes the screen aesthetic, motivating, and inviting? Software screens can be critiqued according to the following visual criteria:

- *Unity* is where each part of the screen is necessary.
- *Balance* creates visual equilibrium on the computer screen.
- *Style* is the combination of distinctive visual features of the screen design which give it a unifying character.
- *Dominance* is what appears as the strongest element on the screen.
- *Repetition* of shapes, patterns, or sizes of elements can unify a screen.
- *Contrast* is where opposites such as light and dark are placed closely together.
- *Arrangement* is the layout of the elements on the screen.

Using visual criteria in the design of software screens leads to professional-looking screens. The application of the visual design elements helps to better achieve educational objectives and promote quality learning experiences.

References

Apple Computer, Inc. (1989). *HyperCard stack design guidelines*. Reading, Mass.: Addison-Wesley Publishing Company, Inc.
Heinich, R., Molenda, M. & Russell, J. D. (1985). *Instructional media*. New York: John Wiley & Sons.
Hlynka, D. (1989). Educational technology: Bullet or butterfly? *AECT-RTD Newsletter*, 13(3), 7-10.
Semrau, P. (1990). *Checklist for good screen design*. Class handout for EDIT 484: Software Authoring Tools in Education. Los Angeles: California State University.

Preparing Computer Education Leadership for the 1990's and Beyond

Robert Hilgenfeld
Gonzaga University, Washington

Abstract

Abstract not available at press time.

Projects
Teacher Education Science Methods Planning Instruction

An Interactive Video Lesson Planning Project for Teacher Education

Charlotte Scherer
Bowling Green State University, Ohio

Abstract

This teacher education project involves the development of a hypercard stack to assist students in planning for instruction. The stack is designed to control the MECC Improving Teacher Effectiveness Laser Disc. It calls up appropriate visual segments as the student progresses through the process of creating a lesson plan, using many of the effective teaching strategies. Although effective teaching strategies are stressed, the lesson plan process in this program does not adhere to a single approach to planning. Rather, the Hypercard stack is designed to facilitate general lesson planning while allowing for incorporation of effective teaching strategies.

The materials will be pilot tested in the spring semester of 1991 with elementary education students enrolled in Computer Utilization in the Classroom courses who need to write lesson plans incorporating the computer as assignments for that class. Research will be done to test for the effectiveness of this lesson plan writing strategy by comparing the quality of lesson plans written using the entire IVD program with groups using just the hypercard stack and with groups writing lessons without the aid of any of the technology. The groups using the technology will also be asked for feedback regarding the use of the media itself. These teacher education students already have an assignment to use the Improving Teacher Effectiveness Laser Disc and Apple IIe programm designed by MECC. In this project they analyze and apply principles that promote student learning. The new Planning for Instruction segment using Hypercard will be a followup to the earlier assignment and will be done at the time in the semester appropriate to the lesson plan assignment.

Both the development process of the IVD program and the preliminary results of the research will be discussed at this session.

One Banana, Two Banana—Science and Math with Integrated Applications

Elizabeth Viau
Montana State University

Abstract

Integrated applications, such as Appleworks or MicroSoft Works, can be taught in the context of scientific investigation. One such project, used with pre-service math and science teachers, uses the scientific method , graphing, and statistics (if students are familiar with statistical analysis) to investigate bunches of bananas at local food stores.

Students develop hypotheses about how many bananas will be in a bunch. Then they go to the store and count inner and outer bananas in a specified number of bunches. Students record the time of sampling and the store where the sample is recorded. They return to class to enter their findings in a data base, and individual data bases are merged. The data is sorted, and summaries are put into the spreadsheet, averaged, and graphed. The students complete the assignment by writing up the experiment in the word processor, including information transferred from the data base and spread sheet. They are required to come up with tentative explanations for their findings, for example, for why banana bunches do not have the same means at different times of the day, and why the means differ from store to store. The ratio of inner to outer bananas can be graphed, and plotting bunch size versus frequency of bunch occurrence yields a normal curve.

A project of this nature allows students to practice the scientific method and the use of an integrated application with materials that are concrete and widely available. The project gives them an opportunity to work with raw data and practice the higher level, problem-solving skills involved in doing science.

Multimedia in Teacher Education

Ronald J. Abate
Cleveland State University, Ohio

Abstract

With funding from the Ohio Board of Regents, the Cleveland State University's College of Education is implementing multimedia instructional materials in its core teaching curriculum. One goal of the multimedia project is to alter the role of instructor from that of lecturer to that of mentor. A second goal is to provide a vehicle for increased student involvement. To achieve these goals four types of multimedia applications have been developed. They include computer controlled videodisc presentations, simulations, teacher tools and a hypermedia database.

- Presentations: The use of videodisc presentations in the classroom has improved student participation by providing undergraduates with shared observational experiences while the addition of computer control over video presentations has increased instructional flexibility. As an indirect result, the videodisc materials have provided a practical reason for coordinating student assignments across courses. For example, classroom management principles observed on videodiscs provided in a reading methods course are then applied in lessons plans created by students in a general methods course.
- Simulations: The courses targeted by the project have an extensive field component. In order to help students with their off campus experiences, tutorials and simulations were developed. These stand alone instructional applications provide an opportunity to practice specific skills prior to attempting these skills in the field. These applications have proven especially appealing to students with little prior experience in a classroom setting.
- Teacher Tools: Computer programs for the production of slides,video, audio and graphics have been introduced and workshops conducted to assist both faculty and students in the creation of instructional materials. Increasingly the focus of the project has moved towards integrating the faculty into the development process.
- Hypermedia: An ongoing effort in the project is the addition of exemplary lessons to a multimedia database. To date, eight lessons have been added to the database with additional lessons presently in development. Each lesson comes complete with text based background information, references, still frame images of student work as well as a video which may be viewed in a linear fashion or accessed via connections in the database. This application has proven especially interesting to our students. It offers a link between what is described in the textbooks and what occurs in the classroom. Although assignments are used to engage the students with the lessons, the database is available to browse through in an open ended manner.

During the past year, this project has enjoyed increased acceptance by faculty who teach the core curriculum. Experiences on how and why the faculty have chosen to implement these innovative materials in their courses will be shared.

Projects
Exciting Projects for Native American Students

A Multimedia Approach to Teaching the Ojibwe Language

Richard Sgarlotti
Hannahville Indian School
Wilson, Michigan

Abstract

The Hannahville Indian School is the only K-12 Indian school in Michigan. It is located on the Potawatomi reservation near Escanaba. Although the tribe has decided that it is important to preserve the Ojibwe language within the community, there are few elders who still speak the language, and only a few students who have seemed interested. However, the students are very interested in computer technology.

The school has been selected as one of the Apple Crossroads Grant recipients, and is using the preservation of the language as one of the focal areas for development of new programs.

This will include the use of video disk technology, with the English language on one audio track, and Ojibwe on the other. It will also involve the use of a Macrecorder, not only to record elders and other native speakers, but also to enable students to record their voices and compare them with the elders'. The audio and video presentations will not only focus on language, but also the culture, including art, music, and history. All of this will be driven by a HyperCard stack, which will also contain scanned images.

The preparation of this will be a cooperative, whole language effort of students and teachers in all subject areas, and community members.

Recreating Native American Legends with HyperCard

George Beekman
Computer Science Department
Oregon State University, Corvallis

Abstract

For the past two summers Oregon State University has hosted math camps for the American Indian Science and Engineering Society (AISES). The OSU/AISES Math Camp provided junior high Native American students with learning experiences in mathematics, computers, and communication. In addition, the camp provided activities designed to help them better understand and assimilate their Indian heritage.

Students had about twenty hours of hands-on computer class time during the two-week camp. In the first week they learn the basics of working with Macintosh computers and HyperCard. During the second week students worked on group projects.

The resulting projects are interactive graphic recreations of traditional Native American legends. Many are animated; some contain sound effects and dialog. In general, the projects look like they were created by seasoned computer users rather that by junior high students with little or no computer experience.

HyperCard's visual authoring environment made it possible for students to quickly learn the fundamentals of computer operation, text editing, graphics creation, and program design while creating impressive and culturally significant projects. The Math Camp experience suggest that tools like HyperCard have potential to transform introductory computer education.

The Native American Educational Network: Its Impact on Teachers and Administrators in Geogrpaphically Isolated Schools

David Basham
Northern Arizona State University

Abstract

The Native American Educational Network (ENAN) originated in 1988 as the Easter Navajo Agency Network. Funded by a grant from the Tandy Corporation, the Bureau of Indian Affairs and the University of New Mexico, ENAN initially served six BIA schools on the Navajo Reservation in western New Mexico. By the fall of 1989, ENAN had linked up as many as 15 schools outside of New Mexico.

Over a three-day period, nearly 70 callers from over 36 locations utilized the network. This kind of network traffic is not remarkable unless one knows where they originated. Lukachukai, Nazlini, Dennehotso, Chinle, Red Mesa, Wingate, Canoncito and Crownpoint are all very remote communities on or near the Navajo Reservation in Arizona or New Mexico. South Dakota is well represented, with callers from Vermillion, Cheyenne River, Agency Village, Marty Indian School and Aberdeen. Networkers also include teachers and students from Idaho, from the Zuni Reservation in New Mexico, Lawrence and Overland Park in Kansas, Muskogee, Oklahoma, and several BIA offices, as well as BIA officials from Washington, DC.

Students and teachers at the University of New Mexico, Navajo Community College, Northern Arizona University and Haskell College also utilize ENAN. Several university courses are offered on the network, but the bulk of usage involves collegial exchange among teachers, student projectsat the rural schools, and administrative coordination among the BIA schools and their Agency administrators. Most readers will not find many of these rural Native American schools on their maps, but for the first time these isolated teachers and students have the means to create and explore the new learning environment provided by computer-mediated communication.

Paper

HyperStories: Cultivating Interactive Reading and Writing on the Macintosh Computer

Joyce Brody and Mary Lange
San Dieso Unified School District, California

Abstract

HyperStories utilizes *HyperCard*, a Macintosh computer application software program, to inspire students to become actively involved with the reading and writing process. The students interact, on the computer, with both stories they create, and the stories of other students.

HyperStories begins with one story line or Story Starter, then branches off into various continuations of the original story. It is a very open-ended program, which has been automated on the Macintosh computer. It is a non-linear program, that allows each reader and writer to make her/his own decisions for stories that branch.The students must incorporate problem solving strategies in both the designing (writing) and utilizing (reading) of a *HyperStory*. Little or no computer programming skills are required from either the student or the teacher to use the *HyperStories* program.

We have been researching and developing applications for the *HyperStories* Project in the San Diego City Schools Deaf and Hard of Hearing program from September of 1988 through June of 1991. The students that have been involved in this project have been between the ages of seven and twenty two years old. Their skill levels have ranged from students who have just recently begun learning the English language, to those students who have been fully mainstreamed in regular high school English classes.

Teachers can easily adapt *HyperStories* to almost any type of classroom, curriculum or student. The students enjoy working and utilizing the writing process via a new medium. *HyperStories* offers a novel and exciting way for students to explore their creativity, while focusing on the writing process.

HyperStories was originally developed to make reading and writing enjoyable for elementary through high school age Deaf and Hard of Hearing students. Our goals are to concurrently motivate and improve the skills of our students in the domains of reading and writing. This program is excellent for promoting reading and writing, not only for the deaf, but for all students! Our experience has shown *HyperStories* to be successful with students who have shown the least enthusiasm for traditional presentations of the reading and writing process.

HyperStories: The Concept

HyperStories utilizes *HyperCard*, a Macintosh computer application software program, to inspire students to become actively involved with their stories. The students interact, on the Macintosh computer, with both their own stories and the stories of other students. *HyperCard* is a software program that allows the user to devise his or her own path for reading and writing. It is a non-linear program, in which the learners make their own decisions. Each experience with the program is distinct. *HyperStories* is an open-ended program, which has automated this non-linear process on the Macintosh computer, and it requires little or no computer programming skills from either the student or the teacher.

We selected the *HyperCard* program for *HyperStories*, because it is a relatively easy program to learn, and it has the capabilities of merging text and graphics effortlessly on a single page. The text, graphics and linking capacities of *HyperCard* enable students to develop and illustrate not only their own stories, but to choose their own sequence for reading and writing the stories. Students can draw their own pictures, utilize clip art, or scan in photographs to illustrate their stories. This experience most nearly duplicates the creation and publication of a real book. Students feel a real sense of accomplishment and pride at the completion and printing of their own *HyperStories*.

HyperStories begins with one story line, or Story Starter. It then branches off into various continuations of the original story. One student originates the introduction of the setting, main characters, theme and plot. Then at some exciting point, the student envisions two or three possible corollaries for the story. S/he passes on the story to another classmate, a student at another school, or perhaps s/he sends the story across the country. Different students then continue the story's various branches, until the story reaches another appropriate point for branching. In this way, the story proceeds to its disparate conclusions.

We decided to develop a schedule to assist the various classes to integrate the writing of *HyperStories*. A Story Starter introduces the story; establishing a setting, characters, theme, and the beginning of a plot. Then at some dramatic point in the story, the student creates three choices or story branches. The story is then passed on to another person or class. As the stories continue, we limit the number of choices, or story branches, and schedule a conclusion to another branch of the story, to keep the stories from becoming unwieldy. The schedule in Figure 1 has proved very successful for our program.

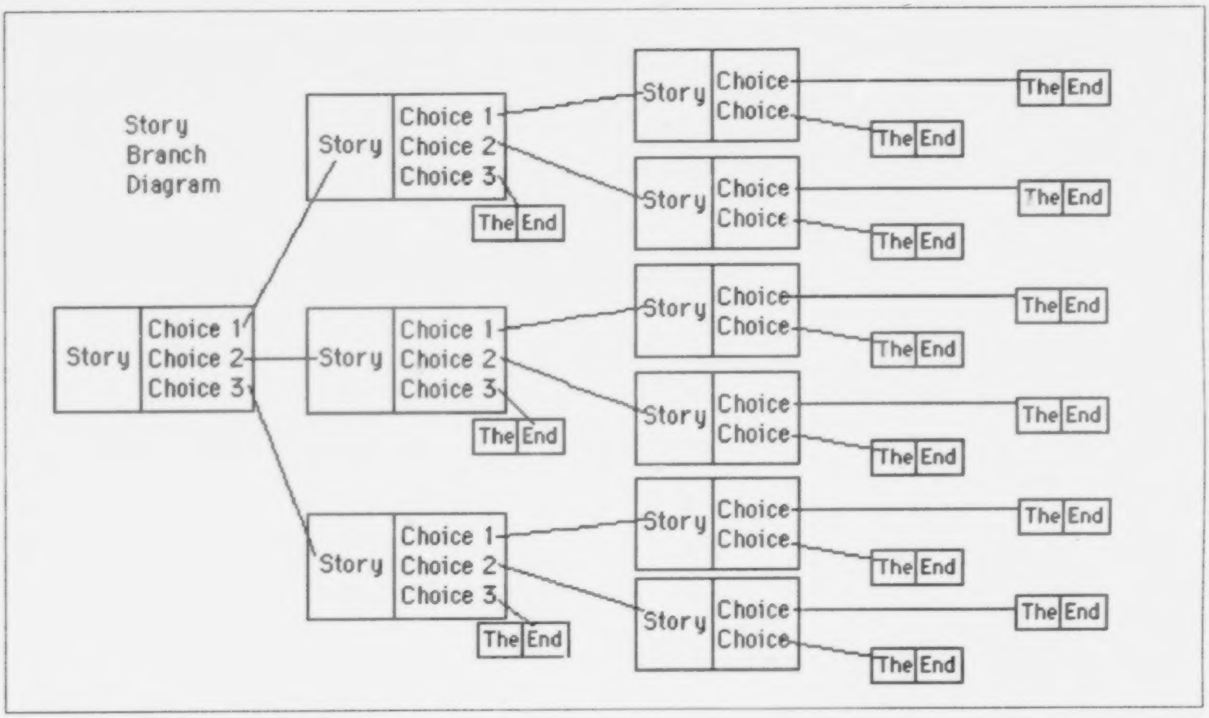
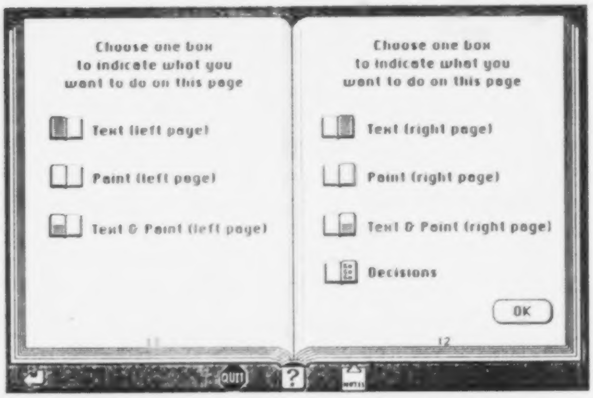


Figure 1. Sample story branch schedule.

HyperStories : Create Your Own

To create a *HyperStory*, a student is constantly involved in a decision-making process. Not only is the student engrossed with the traditional decisions involved with writing an original story, but also with the decisions of the layout for each page of the published product. Decisions are somewhat limited, by the programming of the software, to seven basic choices and any combination thereof. Students can select a pre-programmed page layout by clicking one icon on each page and then clicking **OK**:



Choose one box
to indicate what you
want to do on this page

☐ Text (left page)

☐ Paint (left page)

☐ Text & Paint (left page)

11

Choose one box
to indicate what you
want to do on this page

☐ Text (right page)

☐ Paint (right page)

☐ Text & Paint (right page)

☐ Decisions

OK

12

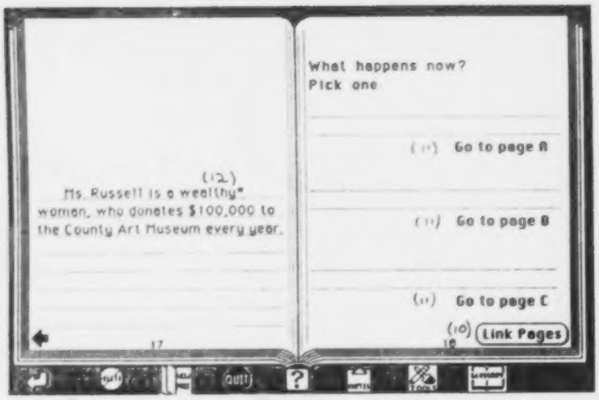
Figure 2. Sample *HyperStories* page layout selection.

The sample on the page below demonstrates two *HyperStory* pages, which have already been selected. One page is set up for Text and Paint (left page). Here a student can draw, scan or utilize clip art to illustrate a story and continue writing the story directly under the graphic. The right page is set up for Decisions (three). A student can envision three possible paths upon which her/his story will be continued by another student (Figure 3.)

Buttons, primarily represented by icons, are located at the bottom of each *HyperStories* page (See Figure 3). Students can click on the buttons and do the following:

Numbers on Figure 3, correspond to the explanations below.

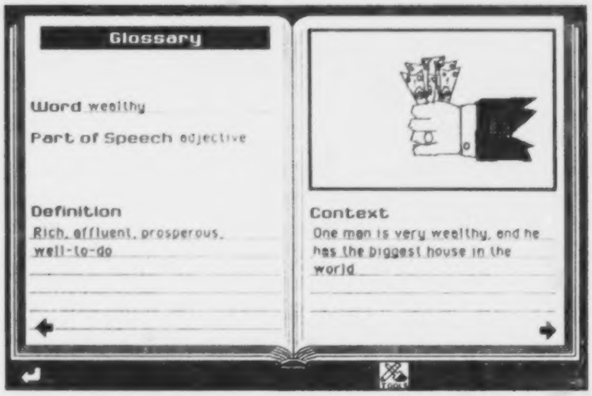
(1) Turns back to previous page.
(2) Turns back to previous story branch.
(3) Deletes the card and returns to the previous page. It will only appear when it is possible to delete a card.
(4) Makes a new choice page. This button also will only appear when it is possible to add a new page.
(5) Closes *HyperStories* and quit *HyperCard*.
(6) Takes you to a credit and teacher options page.
(7) Takes you to a "story ideas" page that gives students a place to organize or summarize their ideas for their stories.
(8) Shows the tool window. You must select the browse tool before closing the tool window.



Ms. Russell is a wealthy* women, who donates $100,000 to the County Art Museum every year.

What happens now?
Pick one

Figure 3. *HyperStories* page and explanation of buttons.



Word wealthy

Part of Speech adjective

Definition
Rich, affluent, prosperous, well-to-do

Context
One man is very wealthy, and he has the biggest house in the world

Figure 4. Sample *HyperStories* glossary page.

(9) Makes a new glossary page, and adds a new word to the glossary, or looks up a word already in the glossary.
(10) Automatically links the number of decisions the student has selected to the appropriate page.
(11) After linking the pages for a decision, the Go to page A, Go to page B, and Go to page C will become buttons that bring you to actual sequential pages.
(12) Asterisked vocabulary word may be entered or "looked up" in the glossary.

The glossary button is a distinctive button in *HyperStories*. It serves a two-fold purpose. It enables students to instantaneously "look up" unknown, asterisked words in the glossary, by highlighting the words and then clicking on the glossary button at the bottom of the page. Similarly, students, at any time, can add their own words to the glossary by highlighting one or more words and clicking on the glossary button at the bottom of the page. Options on the glossary page, include the word's; part of speech, definition, and a sentence with the vocabulary word used in a new context. A space is also provided for students to draw a picture to better clarify the meaning of the word where it is appropriate (see Figure 4).

HyperStories: The Project

The development of *HyperStories* was funded by an Apple Equal Time Education Affairs Grant for the academic years of 1988-90. The Apple grant was awarded to selected applicants who addressed the needs of one of the following groups of students in educational settings; economically disadvantaged, ethnic and linguistic minorities, physically and mentally disabled, or females in math and science.

Our project met the grant's requirements of serving the linguistic minority and physically disabled students of the Deaf and Hard of Hearing program in the San Diego City Schools. The grant was written in collaboration with the Educational Technology Department of San Diego State University (SDSU) and Professor Bernie Dodge. Some of the graduate students from SDSU Educational Technology Department assisted in the development of tutorial stacks to supplement *HyperStories*, Story Starters (the introduction of a story line and the first three possible story branches), and completed *HyperStories* to be used as both reading materials and examples for students.

We researched and developed applications of the *HyperStories* Project in the San Diego City Schools Deaf and Hard of Hearing program from September of 1988 through July of 1990. Approximately 83 Deaf and Hard of Hearing students, in 12 classes, from the second grade through high school have been involved in this project.

HyperStories is versatile, and it lends itself to modifications according to the needs of both the students and the teachers. Students have ranged in ability from those who just recently began learning the English language as well as reading, writing and sign language, to those students who have been fully mainstreamed in regular high school English classes.

In the development of *HyperStories*, we have had particular target areas for improving the reading and writing skills of our students. We have focused on the development of consistent characterization, setting, plot, theme, cohesion and fluency in the writing domain. We also focused on the same facets of reading, while special attention was paid to comprehension skills. It is extremely difficult for a person to successfully continue a story someone else has written, without fully understanding the previous portion of the story. We also have emphasized the need of students to increase both their receptive (reading comprehension) and expressive (written) vocabulary through the use of a glossary page and context clues.

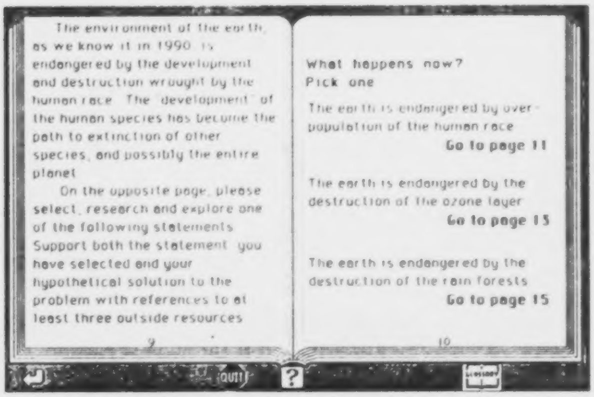
The Process

The first year of the *HyperStories* project was primarily utilized to introduce the Macintosh SE computers, *HyperCard*, and *HyperStories* to the staff and students. It was also used to ascertain their skills in the use of *HyperStories*. We found that the students were far more flexible and open to the use of new technology in the form of *HyperCard* and the Macintosh computers than were the teachers. Many of the teachers were originally dubious about their students' and their own abilities to learn to apply *HyperStories* in an expedient manner. They were extremely concerned that the process (using the Macintosh computer and *HyperCard*) not preempt our true goals of teaching reading and writing. It was also necessary to assist teachers in finding ways to incorporate *HyperStories* into their district mandated curriculum.

We devised a method of introducing the graphics exponent of *HyperCard* to the entire Deaf and Hard of Hearing program that became very popular. In October, all the students drew haunted houses. They used the linking capabilities of *HyperCard* to "open" and "close" doors, windows, and chimneys (by clicking on transparent buttons) to enable an audience to "look inside" and see the witches, ghosts, goblins, and so on, linked by the buttons. The students were thus introduced not only to *HyperCard* and the use of its graphics tools, but they were also surreptitiously initiated into the concept of linking pages through *HyperCard*.

Once the students became comfortable with the graphics, we had them write a linear story using the *HyperStory* Template as a pre-test. Next, students began working within their classes with the story-branching concept. They were then ready to write a *HyperStory*.

Teachers easily adapted *HyperStories* to almost any type of classroom, curriculum or student. It offered a novel and exciting way for students to explore new ideas. In second grade, some students merely drew pictures and either dictated or typed the words to their story. In high school, some teachers utilized *HyperStories* to promote creativity and problem solving techniques. The example below, demonstrates a possible use of *HyperStories* in a high school science class:



The environment of the earth, as we know it in 1990, is endangered by the development and destruction wrought by the human race. The "development" of the human species has become the path to extinction of other species, and possibly the entire planet.

On the opposite page, please select, research and explore one of the following statements. Support both the statement you have selected and your hypothetical solution to the problem with references to at least three outside resources.

What happens now?
Pick one

The earth is endangered by over-population of the human race
Go to page 11

The earth is endangered by the destruction of the ozone layer
Go to page 13

The earth is endangered by the destruction of the rain forests
Go to page 15

Figure 5. Sample *HyperStories* integration into curriculum.

In the elementary classes, students integrated dictionary skills in completing the glossary pages, and in the junior and senior high, the glossary pages provided students with an inducement for using a thesaurus. High school students became very excited about taking an active part in building their own expressive and receptive vocabulary. They were able to choose the words they were interested in learning, from their own creative stories, rather than depending on a teacher to prepare lists of vocabulary words for them to memorize. An interesting ancillary consequence of having the students read and continue each other's stories, was an added incentive to learning new vocabulary words. The students did not want their fellow high school students, particularly younger ones, to know longer vocabulary words than they did.

The Results

In October of 1988, as a precursor to the project, students were asked to write a linear story. Students at the elementary level were shown a variety of well-known books as stimuli. In the junior and senior high schools, the students were simply asked to write a story. Eighteen months later, in March of 1990, the students were asked to repeat this task as a post test.

Extensive research revealed no existing assessments of story development in the specific areas of character, setting and/or plot development. Mary Lange, the second author, designed a three part evaluation for this project.

Students were rated on three different aspects of their stories. The assessments include: 1) word count 2) existence of a story 3) quality of story.

Simply counting the quantity of words in a written language sample is a typical method of assessing the writing of Deaf and Hard of Hearing students. The number of words correlates with the number of propositions. (Yoshinaga, 1983) Stories were also analyzed for story components loosely based on the Stein & Glenn (1979) story grammar analysis.

We predicted that the Deaf and Hard of Hearing students' stories would improve, but minimally through learning and utilizing *HyperStories*. We merely hoped to find a new way of inspiring students to become involved in reading and writing. Based on the evaluations described above, when the scores of pre and post writing samples were compared, there was an average score increase of 73% in overall story development! There was no correlation between student scores and their level of hearing loss. The average increase in word count was 67%. In the pretest, four of the writing samples did not qualify as stories, whereas in the post-test, only one of these was unable to qualify. An average increase of 78% was found in the quality of the stories. (Lange and Dodge, 1991)

The students were surveyed in June of 1990 on their opinions of the Macintosh computer compared to the Apple *IIe*. Results proved that they all enjoyed using both the Apple *IIe* and Macintosh computers; however most students chose the Macintosh over the Apple *IIe* computer. Students chose the Macintosh computer as their preferred tool for reading, writing, and drawing over both the Apple *IIe* computer and paper/pencil/book by at least a two to one margin.

Classroom teachers were highly enthusiastic about the *HyperStories* Project. They noted great improvements in both the quantity and quality of student-written language samples. They noticed the largest improvements in areas of clarity and cohesiveness. They also commented on the motivation students began to exhibit for both reading and writing.

HyperStories : Supplemental Ideas

The *HyperStories* Template can be readily adapted to a variety of age levels, teaching styles, and academic areas. It is basically a "shell" through which students and/or teachers can create. There are two other possible variations we have used with the template. In addition to stories which branch, students can also utilize the template to create linear stories. A simulation could also be constructed, taking advantage of the branching and graphics capabilities of the program. For example, an object could be hidden in a house. The student would need to "move" as s/he searches; in the living room, in the kitchen, and so on to find the object.

Accompanying the *HyperStories* Template are various *HyperCard* support stacks created by graduate students in the Educational Technology Department of SDSU. Character and Plot was developed by Carol Burnett and Gabriel Acosta. Cause and Effect was created by Gabriel Acosta and Joyce Brody. There have also been disks of Story Starters and completed *HyperStories* to promote *HyperStories*. These disks are available through: CUE SoftSwap, (Computer Using Educators).

Conclusion

HyperStories has proved very successful in motivating students to become involved in both reading and writing. Students not only enjoy reading and writing on the Macintosh, but they improve their skills. The writing process is utilized, vocabulary is developed and self esteem is enhanced through the publication of student created projects. Students most "turned off" by traditional reading and writing curriculums flourish using *HyperStories*.

References

Applebee, A.N. (1978). *The child's concept of story: Ages two to seventeen*. Chicago: University of Chicago Press.

Lange, Mary (1990). *HyperStories Report*. Final report submitted to The Apple Equal Time Grant Foundation. Apple Computer, Inc., Cupertino, CA.

Lange, Mary , Dodge, Bernie (1991). *HyperStories: An Interactive Story Writing Environment* , (Unpublished article, in process.)

Scally, Kevin (1984). *The Magic Road: The Story of Red Riding Hood*, London: Mushroom Books Ltd., Grosset &Dunlap.

Stein, N. L. & C. G. Glenn (1979). An analysis of story comprehension in elementary school children. In R. Freddle (ed.) *New directions in discourse processing.*, Norwood, NJ: Abex. [ED 121474]

Story Tree , Scholastic, Inc. P. O. Box 7502, Jefferson City, MO. 65102.

Yoshinaga, C. (January, 1983). *Syntactic and semantic Characteristics in the Written Language of Hearing Impaired and Normally Hearing School-Aged Children.* 206 p. PhD Dissertation, Northwestern University, Evanston, IL. [ED239467 EC161805].

Paper
Managing Multimedia Applications Using DESQview

David L. Doss
Bill E. Swafford
Illinois State University

Abstract

Current hardware technology provides functional microcomputer control of multimedia devices such as CD ROM, LaserDisc, and videoplayers. However, the limitation in current microcomputer operating system technology ofsupporting only a single executing task at a time, prevents full access to these resources. This paper describes how a commercially available windowing system, DESQview, can be used to extend the capabilities of DOS to allow access to these resources through a lesson management program written in an authoring language such as PC Pilot.

Introduction

Currently available technology provides highly functional control of multimedia devices for educational applications. Such devices as CD ROM players can be attached directly to SCSI hard disk controllers, while LaserDisc players can be accessed through a serial communication port. Currently, video players require a specialized interface, but new versions will soon be available that can be controlled through a serial port like LaserDisc players. These devices are typically controlled by issuing commands that establish position on the specific media, followed by a command to begin playing and for how long. The method of issuing commands varies from sending them directly to an output port to which a device is attached with some simple programming language like BASIC, to making requests to device drivers supplied with the hardware or the media through an authoring language such as PC Pilot [Rhodes & Azbell].

Each of these types of media have clear applications to education [Rhodes]. CD ROM players provide access to electronic encyclopedias and other information data bases. LaserDisc players give access to libraries of information and graphics images. Video players allow the display of commercially available video tapes or special tapes recorded locally. The general idea in applying this technology is to use an authoring language to manage the resources and carry on an interactive conversation with a student. Programs written in an authoring language can present options and record student responses.

This paper describes a method whereby all three media types can be combined in a single application using DESQview, a commercial software windowing package. This program supports the execution of multiple DOS programs simultaneously on a single PC. Using this feature, programs that control the video output of the LaserDisc or VCR, can be run concurrently with programs that generate output from the CD ROM player or continue to converse with a student who is using the system. These programs can be active in separate DESQview windows, with a user selected foreground window in charge of the PC display and keyboard. Any window can be selected as the foreground window, at any time.

Equipment Requirements

The DESQview software will work with virtually any true IBM PC compatible microcomputer. However, to obtain an acceptable level of performance in managing the simultaneous execution of several sizeable programs, a hard disk is required. The hard disk can be used to temporarily save unused program storage areas so that the memory occupied by them can be used by other programs. A much better solution is to use an AT class microcomputer with two to three megabytes of extended memory. Accessing extended memory is measured in nanoseconds as opposed to milliseconds for hard disks. Thus swapping programs in and out of extended memory is several orders of magnitude faster than the equivalent operation using a hard disk.

The system monitor should be able to display both text and graphics. This is not required by the DESQview software, but rather by the intended application. The monitor will be used to display the information from the CD ROM player using such facilities as an encyclopedia, which is highly likely to contain graphics images as well as text. The minimal CGA adapter will suffice for many current applications, but the newest software seems to use enhanced graphics. This is a good time to look to the future since EGA and VGA adapters provide CGA emulation modes, and thus will serve well now and for some time to come.

The LaserDisc and VCR will require a video monitor that is separate from the microcomputer console display since they both produce video and audio signals. Both devices can be attached to a single monitor because only one device is in use at a time. Both devices must also be attached to the microcomputer since they will receive control signals from it. The LaserDisc attaches directly to the serial communication port so that no special adapter is required. In the case of the VCR, a

```
1100 '            DISC Operations
1110 OPEN "COM1:4800,N,8,1,CS0,DS0,CD0" FOR RANDOM AS #1
1120 CLS : AA$ = "SA": PRINT #1, AA$: GOSUB 1400
1125 '
1130 '           Get user choice
1133 '
1135 CLS : LOCATE 6, 9: PRINT "VIDEO CONTROL MENU"
1138 LOCATE 9,  10: PRINT "1  Show Single Frame"
1140 LOCATE 10, 10: PRINT "2  Play Video Segment"
1145 LOCATE 11, 10: PRINT "3  Go To Next Frame  "
1150 LOCATE 12, 10: PRINT "4  Pause On          "
1153 LOCATE 13, 10: PRINT "5  Pause Off         "
1155 LOCATE 14, 10: PRINT "6  Stop              "
1160 LOCATE 15, 10: PRINT "7  Fast Forward"
1165 LOCATE 16, 10: PRINT "8  Reset"
1170 LOCATE 17, 10: PRINT "9  Quit"
1175 LOCATE 19, 2: PRINT "Enter number of desired activity -"
1176 IN$ = INKEY$: IF IN$ = "" GOTO 1176
1177 IF IN$ < "1" OR IN$ > "9" THEN GOTO 1135
1178 CHOICE = VAL(IN$)
1180 ON CHOICE GOTO 1505,1600,1610,1620,1625,1630,1640,1630,1660
1183 GOTO 1135
1185 '
1190 PRINT #1, AA$
1200 GOTO 1135
1400 '     SUB TO READ DISC PLAYER
1403 FOR I = 1 TO 200: NEXT     ' TIMER DELAY
1405 INPUT #1, A$
1410RETURN
1500 '      CHOICES
1505 GOSUB 620: AA$ = ARG2$ + "SE": GOTO 1190
1600 GOSUB 660: AA$ = "FR"+ARG1$+"SE "+ARG2$+"PL": GOTO 1190
1610 AA$ = "SF": GOTO 1190
1620 AA$ = "ST": GOTO 1190
1625 AA$ = "PL": GOTO 1190
1630 AA$ = "PA": GOTO 1190
1640 AA$ = "NFNFNFNF": GOTO 1190
1660 AA$ = "RJ": PRINT #1, AA$: GOSUB 1405: CLOSE #1: GOTO 90
```

Figure 1. BASIC Control of LaserDisc

special, and expensive, adapter card must be installed in the PC. It is also the case that the VCR must have two channel capacity, one for addressing and one for video signal. The CD ROM attaches to the PC with a SCSI interface adapter, which is a high performance microcomputer hard disk interface adapter, and therefore a standard commodity.

Software Requirements

In addition to programs supplied with the hardware, such as device drivers, and the DESQview system, three types of software are required: locally developed device control programs, locally developed lesson management programs, and commercial media software platters for the CD ROM and LaserDisc players. CD ROM platters typically come with control programs that provide all access to the data contained on them. LaserDisc platters currently come with a directory that provides addressing information for the data contained on them, necessitating some programming to obtain any desired information. Similar programming is required to establish position and initiate video player output. Such programs can be written in a simple language like BASIC or in an authoring language such as PC Pilot. Incomplete but informative program segments are shown for BASIC control of the LaserDisc in Figure 1 and PC Pilot control of the video player in Figure 2. Complete working programs are available upon request.

As can be seen in these two examples, both the device control and lesson management programming can be performed in BASIC or an authoring language. The segment in PC Pilot looks formidable to the uninitiated, but is actually relatively simple to learn. The advantage in using an authoring language is that much of the tedious detail of screen positioning and response recording is eliminated or at least significantly reduced.

DESQview

DESQview is a windowing system that enhances the functionality of the DOS operating system by providing support for a limited but powerful form of multitasking. The technique of time slicing is used to share the resources of the microcomputer between a single program in the foreground window and multiple programs in background windows. The program in the foreground window controls the keyboard and the console display. However, any background program can be dynamically swapped with the foreground program by the user. A hotkey, the default is ALT, is provided so that such a switch can be requested, or even a new window can be opened.

When DESQview is used with an 80386 based system, there is a companion product, QEMM, that will allow extended memory, memory above one megabyte, to be used as if it were expanded memory in the range 640K to one megabyte. Because of the limitations of DOS, program statements and data must reside in the first megabyte of memory when running. QEMM translates addresses above the one megabyte barrier to addresses below it by using special address registers available on the 80386 chip. This means that programs held in extended memory can be executed by setting a few registers instead of swapping the entire program with programs in storage below one megabyte. Thus there is essentially no overhead in task swapping. This product can be used on an 80286 system that has an expanded memory card.

Windows must be defined to DESQview before they can be accessed. An interactive program is included that assists in defining a window. Such information as full path name of the desired program, window size, whether graphics is required, and whether or not to continue to allow the program to run when it is swapped to the background, must be provided. Also it must be specified whether or not the program can be swapped out if the foreground task needs more memory. Caution is needed

```
R:COLOR.PIL
R:A PC-PILOT PROGRAM ON LANDSCAPE DESIGN

R:THIS SUBPROGRAM INITIALIZES THE TAPE UNIT
R:        Brings in Assembler routine to talk to Tape unit
DX:VIPC$(500)
FX:VIPCPI.BIN
FI:0,VIPC$
R:        If error condition found, message appears.
T(VIPC$(1)=CHR(255)):File VIPCPI.BIN not found!
R:       Change Address for EGA display
C:VIPC$(4)=CHR(002)
C:VIPC$(6)=CHR(003)
EC:

R:     This is the initialization screen.
TS:B0;F15
TX:
:
:
:
:         SYSTEM INITIALIZATION
:               IN PROGRESS
:
:
R:       Address of port for CGA is &H3C3
R:C:P=3C3
R:       Address of port for EGA is &H302
C:P=302
R:       Reset BCD board to start
C:VIPC$(9,80)="OUT   "!!P!!",0"!!CHR(13)
V:VIPC$

R:  THIS SECTION STARTS TAPE UNIT
R:    CHR(13) is carriage return and must be appended to commands
R:     to Tape unit.
C:VIPC$(9,80)="ECOF;TAPE;INIT"!!CHR(13)
V:ViPC$
W:15
V:VIPC$
W: 60
```

Figure 2. Pilot Control of VCR

here. Programs that control devices often do so through the use of an interrupt handler. Such a program should never be swapped out of memory since there is an interrupt vector pointing to it.

After all windows necessary for a particular learning experience have been defined to DESQview, the LEARN facility can be invoked to produce a startup script file that provides the student with a simple means of access. This script file will open all windows, and select the designated window as foreground. At any time, the student can access the other windows using the DESQview hotkey.

An Example

Consider an educational example on developing a research report that utilizes an encyclopedia contained on CD ROM, a word processor, a telecommunications program to provide access to a remote data base, and a PC Pilot program to direct student activity. The script file produced by the LEARN facility of DESQview would open all four windows and select the Pilot program as the foreground window. This program would present instructions to the student concerning the use of the system and give directions on the assignment. The student could look up information in the encyclopedia by using the DESQview hotkey to select that window as the foreground task. The remote data base could be accessed in the same way by selecting the communications window. Using the scratch pad facility of DESQview, the student could copy information from the encyclopedia or communications window into the word processor window to be included in a document or used as research notes.

A ballpark figure for the storage requirements of this configuration using commercially available full function software, is approximately 1.5 megabytes of memory. Thus a two megabyte 80386 based microcomputer should be sufficient to provide a system in which no paging is required.

Networking Multimedia Systems

The expense of a multimedia system makes it highly desirable to consider attaching the system to a network and making it available to all nodes of the network. There are several problems with this approach. The most obvious is that the LaserDisc and video players require a video monitor for their output. There is a reasonably simple solution to this problem. Since the output of these two devices is standard video, it can be made available at all nodes or be routed to a single very large video display if the application is in a laboratory. This suggests duplication of video monitors, but such an expense would be a very small part of the cost of a complete multimedia system.

A not so obvious problem is due to the very nature of this type of system. The software consists of various programs that perform completely separate functions, and were never designed to operate in conjunction with each other. Some of the programs used in a multimedia system control hardware that must be attached to the microcomputer on which they run. However, a standard method of distributing a program across the network is to load the program into the remote node from the network server across the network link, and then run the program on the remote node. This can not work for multimedia applications.

A hopeful way of dealing with this problem is to run the multimedia application on the network server and use a terminate-and-stay-resident (TSR) program [Stevens, Young] on the server to transport the various screen displays across the network and possibly accept keyboard from a remote node. This approach has the apparent advantage of not conflicting with current copyright law since the software involved in the application would always run on the server for which it was licensed. However, as yet only text transmission is available in commercial products. If an application requires graphics no help is currently available.

Bibliography

Quarterdeck Office Systems, DESQview. Santa Monica, CA.

Quarterdeck Office Systems, Quarterdeck Expanded Memory Manager-386. Santa Monica, CA.

Rhodes, D. M. (1987). Open learning in organizations through computer aided interactive video. In V. Hodgson, et al., Eds.., Beyond Distance Teaching: Toward Open Learning. London: SRHE/Open University Press, pp 86-93.

Rhodes, D. M. (1987). Multimediated CAIV: The Botany Program. Interactive Video Showcase, Association for the Development of Computer- Based Instructional Systems, Oakland, CA, November.

Rhodes, D. M. & Azbell, J. (1987). Authoring systems for interactive video instruction. In EURIT 86: Developments in Educational Software and Courseware (Proceedings of the First European Conference on Education and Information Technology). Oxford: Pergamon Press, pp 391-395.

Stevens, A. (1987). Turbo C: Memory-Resident Utilities, Screen I/O and Programming Techniques. Portland: MIS.

Washington Computer Services, PC/PILOT TUTORIAL MANUAL. Bellingham, WA.

Young, M. (1988, March/April). Converting a Turbo C program to a TSR. Programmer's Journal, 6(2), 50, 51, 53, 54, 56-62.

Special Presentation
Computers for Young Children

Harriet Paige
Irvine, California

Abstract

Outline of Presentation for Panel Discussion

Introduction: Speakers first exposure to computers and current use of computers in pre-school setting.

1. The computer as *another* educational tool. Not as a subject to be learned but as a tool to learn other subjects. Computers help children learn:
   - A. Concepts: example-opposites, colors
   - B. Specifics: example-numbers and letters
   - C. Social Skills: example-sharing, interaction with each other
   - D. Learning Skills: example-one thing at a time (Tortise and the Hare)

2. Computers meet needs of both learning types.
   - A. Visual learners have the screen
   - B. Rote learners can repeat the material at their own pace

   Both types can reinforce learning by using multitude of programs on the same subject

3. Helps the teacher organize the day, year, reference materials, and communication of staff, parents, etc..

Closing: The child who made it work.

Special Presentation
The V. I. C. Awards: Video in the Classroom Award Winning Applications

Hall Davidson
President, Video Using Educators
Los Angeles, California

Abstract

Abstract not available at press time.

Projects
Research—Computer Findings

Educational Computing: A Success Story or an Expensive Preoccupation

Andrew Tellep
Pennsylvania State University

Abstract

In light of the ever increasing need to use funds and resources wisely, it is particularly important that decision makers and practitioners have a clear understanding of the potential values of computers in education in order to gain such understanding, one must become informed of the successes and shortcomings of some attempts to improve education through computing and computing systems. This project was an *attempt* to bring together the results of many reports and empirical studies dealing with improvements, or lack of thereof, in education due to educational computing. The project was not confined to any particular level of education and includes results from work done at all levels except graduate level. After reviewing the report prepared at the end of this project, one should have a better view of the types of efforts that have improved education and what has simply been a preoccupation with hardware, software, labs, networks, etc. If our aim is to improve education and not to simply introduce students to computers, are we actually doing things in educational computing that can be show to be educationally sound and that cause positive effects in content areas i.e.—communication skills, problem solving, mathematics, science, social sciences? No matter how this question is answered, the answer is of vital importance to the professionals who will decide the future of educational computing.

University of Northern Iowa Computerized Middle School Curriculum Center

Philip Nelson
University of Northern Iowa

Abstract

Abstract not available at press time.

The Evaluation of Interactive Video Instruction

Sandra Levin
University of Illinois

Abstract

In the past decade, many have expounded the benefits of interactive Video (IV) in education, yet the little empirical research that has been done has come up with mixed results concerning the benefits of IV. This project presentation will briefly demonstrate the use of interactive video to teach earthquake education and will summarize the results of research conducted this past year on the use of IV with five different treatment groups. The results will focus on the effectiveness of IV and its impact on immediate and delayed achievement, motivation and attitudes, and classroom vs. lab setting.

Special Presentation
Logo as Intellectual Laboratory and Vehicle for Self Expression

Gary Stager
Wayne, New Jersey

Paper

Programming Variables: More Than Just a Name

Tronie Gunn
University of Houston, Texas

Abstract

Students do not automatically acquire a mature understanding of sophisticated computing concepts, such as that of variables, simply by experience in writing and editing simple programs. If students have only acquired immature concepts during instruction, their mental models of the underlying system may influence their concept development after instruction ceases. Specific mental models that are useful (such as the "container" model and the "slate" model) in developing the concept of computing variables, as well as specific mental models that are counterproductive (such as the "names" model, the "smart computer" model, and the "black box" model) are characterized.

The concept of variable is both inconstant and elusive. It is hard to describe (Schoenfeld & Arcavi, 1988) and hard to learn (Rosnick, 1981). A survey of definitions of variable reveals that it means different things in different contexts. Understanding the mathematical concept of variable provides the basis for the transition from arithmetic to algebra and is necessary for the meaningful use of all advanced mathematics. In the same way, understanding the computing concept of variable is the key to the production of computer programs that have the desirable property of implementing general algorithms that can be applied to a variety of data.

From the programming point of view a variable is defined as an address or memory location for which a name or label (generally a letter or string of letters and/or other characters, depending on the language) has been designated. The portion of a program in which a given name or label is accessible is called the scope of that variable, although it is more properly the scope of the variable name. Variable names are often accessible throughout an entire program; these variables have "global" scope (for example, all variables in many versions of BASIC). In other programming languages (such as Logo), some variable names may be accessible only within specific procedures and not within other procedures that comprise a single program; these variables have "local" scope.

At any particular time, a variable name will reference a memory location that contains a number (or numerically encoded character string), called the value of the variable, which may change from time to time. Because every memory location contains some bit pattern, and any bit pattern present in a memory location may be interpreted to represent some value, a variable cannot ever be truly "valueless." Some languages distinguish between variables that have explicitly assigned values and those that do not. They identify variables with no explicitly assigned value as "valueless." In many situations, a value for the variable must be designated, implicitly or explicitly, when a variable is created. The value of a variable may change millions of times during the execution of a program. However, when the value of a variable changes, its location, naming, and functional relationships with other elements of the program are invariant.

In many uses of programming variables, the mathematical concept of variables and equality would be sufficient to determine the result of a programming instruction. For example, for assignments that only depend on constant values and/or other variables, a unique value can be associated with the name of the assigned variable. However, the understanding of the incrementation/decrementation assignment makes it necessary to consider a temporal aspect: the value of the variable after the assignment is replaced by the value of the variable before the assignment, plus or minus some value. The student must designate by the same name both the preceding value and the present value (which depends on the former), and has to process the assignment as an asymmetric relation. It seems likely, therefore, that understanding this type of assignment will be extremely difficult for novices.

Subjects of all ages have demonstrated misconceptions related to the concept of programming variables. Some of the most frequent misconceptions have been an assumed relationship between the variable's name and its value or function in the program and misunderstandings related to the passing of parameters and the scope of variables between and within procedures.

All teachers have a vital role in guiding the development of the concept of programming variables and other important computing concepts in their students. But what do teachers know about programming variables after the brief introduction allowed in a typical computer literacy overview course? What mental models might they use to guide the future development of their concept of programming variables after they have completed formal instruction?

The Study

An exploratory study, conducted in the spring of 1990, explored the mental models acquired by preservice and inservice teachers in the context of computer literacy Logo instruction. As part of the study, the subjects' concepts of computing variables were probed in interviews. This report focuses on the results of those interviews.

Methodology

Subjects were 13 student volunteers enrolled in sections of the undergraduate and graduate computer literacy for teachers courses offered at the University of Houston during the Spring Semester, 1990. They participated in the routine course introduction to the Logo computer language with their class sections, completing the same assignments and activities as other students enrolled in the courses. To insure comparability between the Logo experiences of the graduate and undergraduate students, the instructor for the graduate course presented the lecture on variables for both courses. The student volunteers were asked to complete a more detailed experience and demographic questionnaire than that routinely used in the courses. Based on that information, subjects were selected for diversity along several dimensions: (a) classification, (b) gender, (c) teaching experience, (d) programming experience, (e) major field, (f) self-reported computer anxiety, and (g) age.

Subjects were then asked to complete three additional tasks: (a) a booklet in which they were asked to write down in ordinary English "what happens inside the computer when ..." a variety of Logo commands are entered (similar to the booklet developed for BASIC by Bayman & Mayer, 1983), (b) a videotaped interview in which they were questioned about their understanding of "variable" as a computing concept, and (c) a series of brief Logo commands and procedures for which they were asked to predict and explain the results of the commands. This paper will briefly report the results of the first and third tasks, while focusing on the second. Questions asked in the "variables" interviews included the following (with additional clarifying probes, as needed):

1. What is a variable?
2. Does every variable have a name?
3. Give an example of a name which a Logo variable could have.
4. How many names can a given variable have? Could a variable have more than one name at any particular time? Why?
5. When and how does a variable get a name?
6. Can the name of a variable be changed? If so, how?
7. Does every variable have a value? Explain.
8. Give an example of a value which a Logo variable could have. Give an example of a value which a Logo variable could not have.
9. How many values can a given variable have at one time? Explain.
10. When and how does a variable get a value?
11. Can the value of a variable be changed? If so, how?

Results: The Logo Booklet

Three categories of language statements that were studied included direct assignments (using MAKE), output (using PRINT), and repetition (using REPEAT). The subjects tended to verbalize conceptions of the statements that either failed to include the main idea or that included outright misconceptions for all three categories of statements. For example, the location in which a value obtained by incrementation would be stored (replacing the variable's previous value) was not clear and subjects commonly indicated that all assignment statements also produced some form of visible output. Subjects often failed to mention the repetition when trying to describe the repetition of an assignment than when trying to describe the repetition of printing. Although all subjects had demonstrated "hands-on" competence in using Logo, this experience was not sufficient for the mature understanding of programming.

Results: The Logo Prediction/Explanation Examples

The errors and misconceptions observed in the prediction/explanation portion of the study were of several types: incomplete/incorrect knowledge, apparent inconsistencies of the Logo environment, and attribution of human-like reasoning capabilities to the computer. Many errors were similar to those observed in the Logo booklet transaction lists. These include a variety of difficulties with assignments (especially incrementation/ decrementation) regarding related output and their use in repetition.

Of the errors not obviously related to the transaction lists, perhaps the most common were those that attribute to the computer human-like will or understanding. Subjects who suggested that the actions of the computer were not within programming control or that the computer could understand the names of procedures and/or variables and act accordingly evidence counterproductive mental models. High anxiety subjects, subjects whose major fields were not mathematics or science, and subjects with no programming background were the most likely to make these errors.

Results: Transactions and Predictions

There was a statistically significant Pearson correlation of $r = .68$ ($p < .05$) between the number of incorrect/incomplete conceptions in the Logo booklet transaction lists (possible misconceptions) and the number of examples presented before no later examples were predicted incorrectly. This is equivalent to the correlation between the number of correct conceptions and the latest number of consecutive prediction examples that were at least partially correct.

Results: The "Variable" Interview

Subjects generally exhibited relatively incomplete concepts of variables. Several subjects demonstrated some mixing of the ideas of variables and procedures in Logo. Results of the responses to questions in the interview are summarized in Table 1.

Table 1.
Components of a Mature Concept of Variable Observed in "Variables" Interview

Variables have:	Optional	Required
* a name	100.0%	83.3%
* a value	100.0%	58.3%
* a location	33.3%	33.3%
* a scope	0.0%	0.0%

Variable characteristics:	Name	Value
* can be changed	83.3%	100.0%
* only 1 at a time	75.0%	91.7%
* valid example	91.7%	91.7%

Assignment by:	
* MAKE command	16.7%
* TO command	16.7%
* other or unknown command	58.3%

The subjects generally presented naive but reasonable descriptions of variables as placeholders or substitutes (similar to the mathematical concept of variable). They identified some defining characteristics of variables but omitted others. For example, all agreed that variables could have names and values and most indicated that variables were required to have both. Only a few suggested that variables could have a location, but those subjects indicated that a location was a requirement. No subjects brought up the characteristic of scope (not particularly surprising for novices!). It is the lack of inclusion of location as a critical, required component of a computing variable that may have been responsible for many of the observed misconceptions.

The subjects generally agreed that variables could have only a single name and a single value at any time. They could usually offer valid examples of both names and values. They agreed that the values of variables could be changed, although they were usually unable to specify the command for doing so. Most also said that the name of a variable could be changed, which is not possible within Logo. Given the commonly identified mechanism for accomplishing this change (editing a command or procedure), they may have meant that the name of a variable was arbitrary and subject to the programmer's choice. The subjects were generally uncertain about what creates a "new" variable and what happens to "old" variables when they are no longer in use.

Mental Models

"Mental models are the mechanisms whereby humans are able to generate descriptions of system purpose and form, explanations of system functioning and observed system states, and predictions of future system states." (Rouse & Morris, 1985, p. 7) Although the variables interview was intended to probe the subjects' concepts of programming variables at that time, the subjects' mental models of variables (and of computing in general) would likely influence the future development of those concepts after the subjects are removed from the guidance of a course instructor.

Subjects seldom articulated a clear picture of the mental models they may have been using to make their predictions and explanations. However, based on the patterns of errors and misconceptions, it is possible to infer broad characterizations of the mental models that may have been used. No claim is made that these models are the only ones that could have been in use.

Several mental models that may have been employed did not relate directly to variables. Instead, they related to the subjects' overall view of computers and problem-solving. These include (a) the "smart computer" model, in which the computer or the currently executing program is expected to possess intelligence, understanding, and will; (b) the "black box" model, in which the functioning of the computer is considered mysterious and not easily comprehensible; and (c) the "use everything" model, in which problem solving is viewed as simply combining as many as possible of the apparent facts into the "right" form. These models are highly counterproductive, directing the attention to features that are only coincidentally related to the actual results. Some subjects may have used combinations of these models, no models at all, and/or other, more variable-specific models.

Some of the possible mental models used related directly to the subjects' concept of computing variables. These models include the "container" model, the "slate" model, the "names" model, and the "procedures" model. Each of these is explained in more detail below.

The "Container" model. In this model, subjects view variables as named containers for values. This type of model is among the most commonly suggested models for variables in textbooks and typical programming instruction (e.g., Fox & Waite, 1981; Luehrmann & Peckham, 1983). Programming instruction typically stresses that these containers (variables) can only hold a single value at a time. However, students may imagine that the containers (variables) can hold many values at once, just as the real-world containers with which they are familiar can hold many things at one time (DuBoulay, 1989). Although imagining potentially empty containers in a situation such as Logo programming that does not allow empty variables to exist is unlikely to cause difficulties in making predictions or explanations, it is another possible shortcoming of this model.

An important potential shortcoming of this model becomes obvious when the possibility of both local and global variables is considered. To account for the possibility of both local and global variables, it is necessary to allow for multiple sets of containers to exist. The containers within each set would be required to have unique names, but containers in two different sets could have identical names, because the set to which the container belonged would be a distinguishing identifier. There also would be one set designated as the "global" set, which would become, in effect, joint property of all the "local" sets of containers and would be referenced if no "local" container had an identical name.

One subject who consistently referred to an analogy of variables with boxes obviously used a containers model. This model was clearly useful for her. Other subjects may have also used this model.

The "Slate" model. This model was not directly observed as the only possible explanation of the predictions of any subjects, but may have been the underlying model of many successful subjects. In this model, computer memory is viewed as containing an erasable slate on which the names and current values of variables can be written. A memory reference copies the contents from the slate, while an assignment writes a new value, after erasing the previous one, if any. Subprocedure calls can be correctly viewed as not simply passing control to a new set of commands (as in BASIC), but also making a set of variable assignments based on the current values of specified variables (or constants). Subjects with this model would have no difficulty with multiple assignments - the slate can only have one value (the latest assigned) for a given variable.

The "Names" model. Subjects with this model seem to view variables as location-less abstractions, similar to mathematical variables. These variables can have many values, but only one at a time (for a given exercise). However, the value is linked to the name and the name is globally known (at least for the duration of that exercise). This model seems to be basically parallel and static. A temporal dimension is definitely missing.

This model causes severe problems in encounters with incrementation/decrementation assignments. It is not reasonable to consider that C can in any way be or become C+1. Though Logo uses a command (MAKE) that suggests the temporal nature of the assignment, instead of the potentially more confusing equal sign used in some languages, subjects using this model were still probably confused.

Another error that may have come from this model involves difficulties with repeated assignments. Making two incompatible assignments (such as making TIMES both 15 and 25) is incredibly confusing when processed through the names model.

This model seems unlikely to evolve into a more adequate model without knowledgeable intervention. Subjects have a wide range of examples in mathematics upon which to base this model. The limited exposure to programming in an overview course is unlikely to offer sufficient examples to discard this model, which worked successfully in other contexts. The instructor must clearly present and explain examples for which this model is insufficient and misleading.

The "Procedures" model. Subjects with this model may have overgeneralized their models of procedures in Logo to use them as models for variables. This could result in the assumption that incrementation/decrementation assignments would create new copies of the variables named in a manner similar to recursion of procedures. As there is no command-based equivalent of in-place replacement of a procedure in Logo, generating an explanation of this type of command could be difficult from this model. Once a procedure is defined, the definition is stored ("remembered") and is later invoked by naming it. This is similar to the expectation of several subjects that assignment equations are remembered and can reproduce the specified action later by name. Object Logo does not allow the user to define a procedure with the same name as one already defined in the workspace; an error message results. Some subjects suggested that repeated assignments would also result in an error message or that an assignment to a variable called W might cause a problem because of the already defined procedure named W.

Conclusions

Subjects did not exhibit mature concepts of programming variables, although they typically did not evidence strong inaccuracies in their naive concepts. Given appropriate time and opportunities to tune their developing concepts, the subjects could be expected to develop mature concepts of programming variables. If most will not have additional formal instruction regarding programming variables, their mental models (if they have formed one) could be expected to guide any further development of the concepts of programming variables.

Subjects with "smart computer" and "black box" models could be expected to have limited future development of their concepts of variables without further instruction. Their mental models of computing lead them to feel that they cannot control or understand what happens, so they make little effort to do so.

Both the "container" and "slate" models (and to a smaller degree, the "procedures" model) usually support good predictions about the functioning of Logo variables. Instructors could be encouraged to teach these models of variables, with their limitations. Experience with the Logo environment in particular and with computing in general might sufficiently identify and clarify limitations of each model (without further formal instruction) so a mature concept of programming variable can result (perhaps without the component of scope, unless specific knowledgeable intervention occurs).

Subjects utilizing the "names" model seem to be doomed to experience confusion and frustration. Subjects can readily construct this mental model based on years of exposure to it through mathematics. Unfortunately, the fundamental natures of mathematical variables and computing variables are different! Mathematical variables are abstractions - they can only be identified by their names. But computing variables are more than just a name! Computing variables have a location - the name is really only a convenience for the programmer. Computing variables with the same names can assume different functions within the context of the same program and computing variables with different names can communicate values from one part of a program to another. Care should be taken to eliminate this model from the repertoire of all computing students, but especially those who plan to become teachers and who might pass this model on to other students.

Bibliography

Bayman, P., & Mayer, R. E. (1983). A diagnosis of beginning programmers' misconceptions of BASIC programming statements. Communications of the ACM, 26, 677-679.

DuBoulay, B. (1989). Difficulties of learning to program. In E. Soloway & J. C. Spohrer (Eds)., Studying the novice programmer (pp. 283-299). Hillsdale, NJ: Lawrence Erlbaum.

Fox, D., & Waite, M. (1981) Pascal primer. Indianapolis: The Waite Group, Howard W. Sims.

Luehrmann, A., & Peckham, H. (1983). Computer literacy: A hands on approach. Mission Hills, CA: Glencoe/McGraw-Hill.

Rosnick, P. (1981). Some misconceptions concerning the concept of variable. The Mathematics Teacher, 74, 418-420.

Rouse, W. B., & Morris, N. M. (1985, May). On looking into the black box: Prospects and limits in the search for mental models (Report No. ONR-TR-85-2). Arlington, VA: Office of Naval Research. (ERIC Document Reproduction Service No. ED 268 131)

Schoenfeld, A. H., & Arcavi, A. (1988). On the meaning of variable. The Mathematics Teacher, 81, 420-42

Paper

Teaching the Idealizations: Altering Science Alternative Conceptions with Microcomputer Simulations

Herman G. Weller
Texas Tech University

Norman R. Dodl
Virginia Tech

Abstract

A large part of the science student's prior knowledge, which has been derived from the individual's previous experiences, is not compatible with what is taught in science classes. The naive models of the natural world that the child brings to the classroom are often quite resistant to traditional teaching. If both the teacher and the student cannot detect and change these naive models, or *alternative conceptions*, the student's ability to continue learning science may be seriously impeded.

This is a report on the development and investigation of two microcomputer-displayed simulations of *idealized scientific phenomena* that were designed to facilitate the 8th-grade science student's alteration of three Aristotelian alternative conceptions. The instructional strategies incorporated into the computer simulations were consistent with a theory of science conceptual change that: a) facilitated the student's recognition and discovery of a phenomenon that was contradictory to his or her conceptual framework and that was typical of the corresponding scientific concept, and b) allowed the student to manipulate the objects and relationships of the phenomenon, experiencing the consequences of that action, so that the student would adjust his or her conceptual categories until the phenomenon became anticipated.

Eighth-grade students who were selected for possession of alternative conceptions were helped by the computer simulations to alter their naive conceptions to a significant degree. This was shown by a post-test administered immediately after the students had worked through the simulations, and by a retention test administered to a sample of the students 1.5 months later.

Introduction

For many children, science is a difficult subject to learn. A large part of the child's prior knowledge, which has been derived from the world of everyday experience, is not compatible with what is taught in science classes. The naive models of the natural world that the child brings to the classroom are often quite resistant to traditional science teaching. If both the teacher and the student cannot detect and change these naive models, or *alternative conceptions*, the student's ability to continue learning science may be seriously impeded.

One solution to this instructional problem would be to construct a microcomputer-based system that a teacher could use a) to diagnose a student's possession of alternative conceptions, and b) to help the student develop a conception more in line with the current scientific viewpoint to replace each alternative conception. This is a report on the development and investigation of the remediation component of such a system: two microcomputer-displayed simulations of *idealized scientific phenomena*, which place the student in environments where a) he or she can *play the game of science* in contrived situations where his or her naive conception is at conflict with the displayed phenomenon, and b) he or she can experiment with examples of the more "scientific" concept until the conflicting phenomenon becomes the anticipated phenomenon.

Alternative Conceptions

For over twenty years, science education research has been accumulating evidence that a significant number of science students of all ages, weak as well as strong learners, possess alternative conceptions (Hasweh, 1988). These are the alternative views of the world that have been derived from the individual's previous experiences. They often persist in spite of several years of traditional instruction in concepts that are more in line with the scientist's viewpoint and that contradict the naive conceptions. In many cases, these naive models of the natural world have been found to occur regularly across different students. In the area of force and motion, many present-day alternative conceptions have been found to be similar to early explanations (Hasweh, 1988).

The students' possession of alternative conceptions of force and motion should not come as a surprise to the science educator, for the world of force and motion in everyday experience is not identical to the science models of the classroom that have been abstracted from the natural world. The world of everyday experience is indeed largely an Aristotelian world (Garrison & Bentley, 1990). For example, the child's world outside of school is not airless, frictionless, or without gravity. Heavier objects do fall more quickly than lighter ones (say, a stone and feather falling through the air). Sliding objects do require forces to keep them in motion (say, a sled on lake ice) and the retarding friction is not often recognized as a force.

It is often in the physical science course in the eighth or ninth grade that many students first encounter extensive formal instruction that contradicts several of their naive conceptions of force and motion. In spite of the instruction there and in the grades following, the same alternative conceptions are also held by secondary and university students. For example, Champagne, Klopfer, and Anderson (1980) found that most of the college students they surveyed believed that objects fall at a constant speed, a speed that the students thought (as had Aristotle) to be proportional to the weights of the objects. Because the eighth-grade confrontations between students' naive dynamics beliefs and instruction on the scientific viewpoint have significant importance for the children's future science learning, the present investigation concentrated on fostering conceptual change of eighth-graders who were found to possess Aristotelian alternative conceptions of force and motion. These were eighth-graders who were found to hold one or more of the following three alternative conceptions several weeks after their physical science classes had covered force and motion.

1) *Quickness of fall depends on mass* conception
   With no air resistance, the downward movement of a more massive object is faster than that of a less massive object.
2) *Unbalanced force* conception
   If an object is moving and no unbalanced force is acting on it, the object will come to rest.
3) *Direction of vertical movement depends on type of material* conception
   a) Light objects rise because they are made of material that naturally tends to move upward;
   b) heavy objects fall because they are made of material that naturally tends to move downward.

Physical Science: An Alien World of Abstractions?

Most students, and too many science teachers, do not realize that physical scientists have commonly dealt with the staggering complexity of the natural world by constructing idealized models of various real-world phenomena. Hence, one typically finds such statements in eighth-grade science as 1) "When objects of different mass are dropped simultaneously from the same point *in a vacuum*, they strike the ground at the same time" and 2) "Objects will continue moving indefinitely unless they are affected by some *outside influence*." However, in his or her experience, the student has not dropped objects *in a vacuum* and has not moved objects where they were unaffected by *outside influences*.

Unfortunately, many science teachers (and perhaps some physical scientists) believe that the abstract models of science determine the natural world, rather than follow from it. Certainly, for many students, the *network of idealizations* that is physical science, where one explains the natural world by referring it to mathematical formulae, is indeed an alien world with an alien language. In the authors' view, a theory for changing a student's naive conceptions in science should take account of the fact that the corresponding scientific conceptions are often abstractions from the world of practical experience, and should allow for student practice with examples of the appropriate idealized scientific situation.

A Model for Changing the Science Student's Conceptions

Hasweh (1988) has contended that a conceptual change study should rest upon the foundation of a specified, appropriate theory of conceptual change. Instructional strategies can then be proposed and implemented in order to test the theory.

The first author (Weller, 1990) has constructed a theory for instructionally eliciting a student's conceptual change in science. The crux of the theory is that the science student who has an alternative conception can be helped to discover a phenomenon that is contradictory to his or her conceptual framework and that is typical of the relevant scientific concept. The teacher can accomplish this by allowing the student to examine the phenomenon repeatedly, in effect *playing* with examples of the scientific concept by acting upon the objects and relationships of the phenomenon and then experiencing the consequences of that action. The result is that the student gradually adjusts his or her conceptual categories until the phenomenon becomes anticipated.

For the construction of this conceptual change theory, Weller (1990) has drawn heavily upon 1) the theoretical views of Thomas Kuhn, both on the role of anomaly in the discovery of a new and unsuspected phenomenon (Kuhn, 1970) and on the function of *thought experiments* in science (Kuhn, 1977), and 2) the thoughts of John Dewey (1916) on the relation of experience to thinking. The conceptual change theory has drawn also upon a) the model for the facilitation of conceptual change in science that was proposed by Posner, Strike, Hewson and Gertzog (1982), and b) the criticism of the latter model by Garrison and Bentley (1990) for its assumption that a student's learning is a *rational* activity, and for ignoring *prerational* considerations in learning.

Weller (1990) has described an individual learner's process of conceptual change in terms of the three stages that Kuhn (1970) has portrayed in a scientific community's discovery of a new and unsuspected phenomenon: the practice of normal science, awareness of anomaly, and conceptual adjustment. To help the student become aware of anomaly, Weller has employed Dewey's (1916) notion of an individual's learning experience as an active-passive affair, in which "Experience as trying involves change, but change is meaningless transition unless it is consciously connected with the return wave of consequences which flow from it." The student learns when he or she acts upon something and then undergoes the consequences of that action.

Posner et al. (1982) have proposed that for radical conceptual change to occur the learner must be dissatisfied with his or her existing conception and the new conception must be intelligible, initially plausible, and apparently fruitful for future use. Garrison and Bentley (1990) have maintained that Posner's description is incomplete because learning for the beginning science student takes place before the individual has formed complete "scientific" concepts and categories. To this student the idealized system of science represents another world, with standards of what constitutes valid reasoning and sound evidence that are entirely different from the world of everyday practice.

Weller (1990) has listed the following as necessary characteristics of the instructional situation that presents the contradictory phenomenon to the student:

1. It presents the scientific idealization of the contradictory phenomenon.
2. It exhibits cues that the student usually employs when making judgments in this area.
3. It confronts the student with a contradiction.
4. It provides the student with enough control over the situation that he or she can restart the contradictory phenomenon several times.
5. It omits distracting variables that are not important for understanding the anomalous phenomenon.
6. The conflict deduced by the student from the contradictory phenomenon must be one that the student could confront in experiencing nature.
7. The deduced conflict must be one that, however vaguely perceived, has confronted the student before.

The Microcomputer Simulations

Simulation is the "process of interacting with a model that represents reality" (Lunetta & Hofstein, 1981). According to Gagne (1962), a simulation a) represents a real situation in which operations are carried out, b) provides the user with certain controls over the problem or situation, c) omits certain distracting variables that are irrelevant or unimportant for the particular instructional goals.

There have not been many rigorous studies to date of the validity and effectiveness of computer simulations in physical science instruction. Twelker (1968) has suggested that the degree to which a simulation corresponds with the real-world situation it is designed to simulate may be crucial to learning. He felt that if there is a credibility gap between instruction and the operational world, then the learner may be at a disadvantage when it comes to performing in, and/or understanding, the real world. This view does not take into account the fact that the world of science is actually an abstraction from the world of practical experience.

In an alternate view, the authors maintain that for a simulation designed to induce conceptual change, strict adherence to the conditions of a real-world situation is much less important than it is to present the scientific idealization of the real-world situation wherein a phenomenon obviously contradicts the student's naive conception(s). Such an idealized situation strongly resembles the *thought experiment* of the scientist. A *thought experiment* is one that is imagined by the scientist but not actually carried out. It may involve a situation that has not been examined in a laboratory, or even a situation that could not possibly occur in nature (Kuhn, 1977). A well-known example is Galileo's *thought experiment* where he investigated the dependence of an object's acceleration on its weight by imagining two small equal weights falling separately side by side or chained together. Weller (1990) holds that the conditions necessary for facilitating a student's conceptual change with a simulation are the same as the conditions proposed by Kuhn (1977) as being necessary for the effectiveness of *thought experiments* in eliciting conceptual change.

Arguing that from thought experiments most people learn about both their own conceptions and the world, Kuhn (1977) held that a thought experiment must be capable of confronting the thinker with a paradox of such impact concerning the person's incorrect conceptions, "transforming felt anomaly to concrete contradiction," that the person is both informed that he or she is wrong and provided with clues necessary for *reforming* the conceptions. Concerning the conditions necessary for this impact, he concluded first that although the situation imagined in the thought experiment need not even be potentially realizable in nature, it must exhibit the cues which the thinker usually employs when making judgments in this area. Secondly, the conflict deduced from the imagined situation must not only be one that nature itself could present, but must also be one that, however, vaguely perceived, has confronted the thinker before.

The two computer simulations in the present study were designed to facilitate the student's conceptual adjustment in the following ways. Each simulation focused the student's attention on a phenomenon that was a scientific ideal case and that was in contradiction to one or more of his or her diagnosed alternative conceptions. The student reran the first simulation ten times and the second simulation five times, in order to have time a) to become aware that the highlighted phenomenon was contradictory to his or her conceptual framework and b) to adjust his or her conceptual categories until he or she came to expect the phenomenon each time. Each simulation exhibited cues by which the student usually judges the motion of objects. Because the student's conceptual change might be accompanied by student resistance, each simulation was designed to abate many feelings of resistance to discovery by employing appropriate principles of human-computer interactivity (Weller, 1988) to a) make the student feel in control of the simulation, and b) provide him or her enjoyment and satisfaction in running the simulation.

Objects Falling/Rising simulation

For the *Objects Falling/Rising* simulation (Figure # 1), the student chose the materials from which two objects were composed and the height on a tower from which those objects fell. The student also chose the mass of each object, unless the

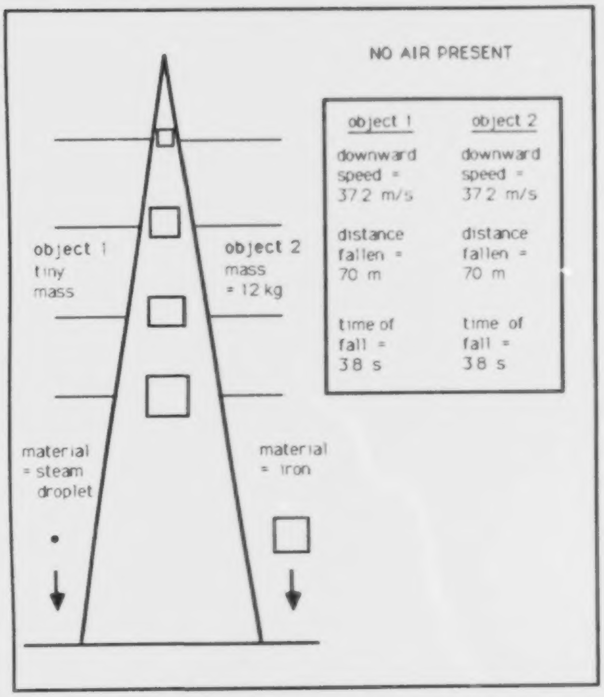

NO AIR PRESENT
object 1
downward speed = 37.2 m/s
distance fallen = 70 m
time of fall = 3.8 s
object 2
downward speed = 37.2 m/s
distance fallen = 70 m
time of fall = 3.8 s
object 1
tiny mass
object 2
mass = 12 kg
material = steam droplet
material = iron

Figure 1. The *Objects Falling/Rising* simulation. Two objects fall or rise on opposite side of a tower. Air is alternatively present or not present in successive runs of the simulation.

object was a steam droplet or a dust particle (whose masses were merely stated as "tiny"). The simulation runs were alternately in an environment of *air resistance* or *no air resistance*. In the cases when the objects fell from the tower to the ground, the right half of the screen showed at different times a) no supplemental information, b) two downward arrows representing analogously the current speed of fall of the objects, c) the increasing times of fall, speeds, and distances fallen of the objects. In the cases when the objects did not fall (i.e., in the case of the steam droplet or the dust particle in an environment of air resistance), the supplemental information was not provided.

When the objects fell with no air resistance, they moved at opposite sides of a tower and struck the ground at the same time. Both the parallel fall of the objects past tower stories and their simultaneous impact with the ground presented cues that a person customarily employs when making judgments about falling objects.

The *Objects Falling/Rising* simulation was designed to provide, in the case of no air resistance, a situation in direct conflict with the beliefs of the student who had a *quickness of fall depends on mass* alternative conception (i.e., who believed that, with no air resistance, the object with more mass will strike the ground first). The conflict was one that can occur in nature when objects much denser than air are dropped within a few meters of the earth, and one that the student had quite likely encountered before (even if only vaguely perceived). The simulated fall of the two objects, as well as the supplemental information, provided clues to allow the student to work toward altering his or her conception.

The *Objects Falling/Rising* simulation was also designed to provide, via the contrast of the air-resistance and no-air-resistance environments, situations in direct conflict with the beliefs of the student who had a *direction of vertical movement depends on type of material* conception (i.e., who believed that light objects rise because they are made of material that naturally tends to move upward, and that heavy objects fall because they are made of material that naturally tends to move downward).

Cube Moving simulation

For the *Cube Moving* simulation (Figure # 2), the student chose the mass and initial horizontal velocity of a cube. For each simulation run, the computer system chose four new forces that were exerted on the cube throughout its motion (an upward force, downward force, force from the left, and force from the right). While the cube moved across the screen, the forces upon the cube were represented by arrows with lengths proportional

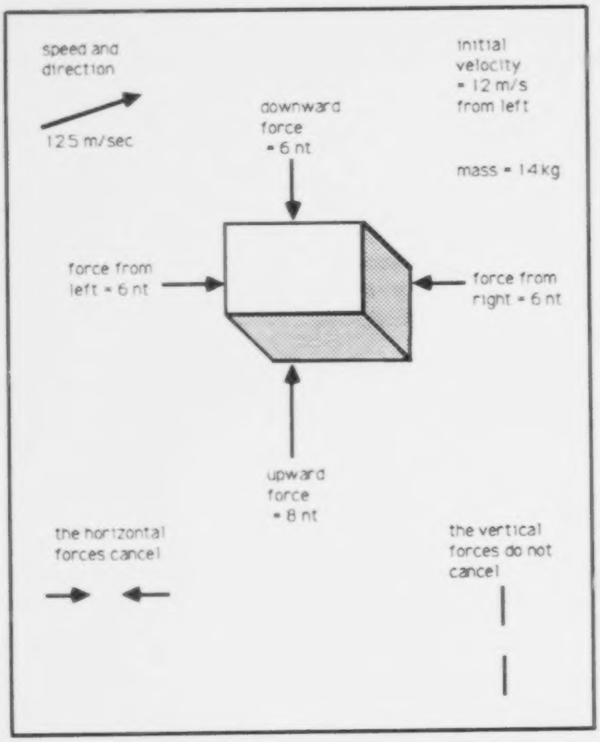

speed and direction
12.5 m/sec
downward force = 6 nt
initial velocity = 12 m/s from left
mass = 14 kg
force from left = 6 nt
force from right = 6 nt
upward force = 8 nt
the horizontal forces cancel
the vertical forces do not cancel

Figure 2. The *Cube Moving* simulation. As the cube moves, the incident forces and the current speed and direction are always shown. After the start of the simulation, the initial velocity, mass, and force-cancellation diagrams disappear.

to their magnitudes. During the motion, an arrow in the upper left portion of the screen represented the current velocity of the cube. The magnitude of the velocity was printed alongside the arrow.

Before the *Cube Moving* simulation started, two diagrams at the bottom of the screen informed the student whether the horizontal forces on the cube were balanced and/or whether the vertical forces were balanced. The motion of the cube relative to the screen borders, and to the cube's initial starting point, presented cues that a person typically uses when making judgments about moving objects.

Whenever the four forces were balanced in the *Cube Moving* simulation (i.e., when the resultant force was zero), the cube did not come to rest. Instead, it continued to move with its initial velocity. This situation was in direct conflict with the beliefs of the student who had an *unbalanced force* conception (i.e., who believed that a moving object on which the forces are balanced will come to rest). The conflict was one that could occur in nature when massive objects moved across low-friction surfaces (such as ice), or when objects moved at large distances from the earth (which the student might have observed before in films of astronauts on board a satellite). It was also a conflict with which the student had probably been confronted before. As with the objects in the *Objects Falling/Rising* simulation, the motion of the cube, as well as the supplemental information, provided clues to allow the student to work toward altering his or her conception.

Methodology

A series of interventions was implemented with a representative sample of students drawn from eighth-grade physical science classes at two schools, with 29 and 31 students, respectively. The first school was a suburban private school for grades K-12; the second was a high school that drew students from a large rural area containing several large towns. The 60 students from these two schools had studied force and motion in their science classes two and six months earlier, respectively.

When students were detected by a pre-test as possessing alternative conceptions, the system presented the two simulations, which were designed to facilitate the student's alteration of one or more of these naive conceptions. The control group students did not do the simulations. They filled out questionnaires concerning their attitudes and opinions about the pre-test, while the experimental group students worked through the two simulations. We determined evidence of immediate change by administering a post-test both to students who had worked the simulations and to students who had not. We assessed evidence of the students' retention of simulation-elicited changes 1.5 months later by readministering the post-test to a sample of those students who had worked the simulations.

Results and conclusions

The students who were selected for possession of alternative conceptions were helped by the computer simulations to alter their naive conceptions to a significant degree. A post-test supplied evidence of the students' short-term conceptual change. The students in the experimental group answered an average of 5.9 questions correctly (out of eight) on the post-test, with a standard deviation of 1.5 questions; the control group students answered an average of 4.8 questions correctly, with a standard deviation of 1.8. A *Students t test for independent samples* showed that the experimental mean was significantly superior to the control mean ($t=2.64$, $p=0.005$).

The retention test that was taken by a sample of seven students 1.5 months later supplied evidence of robustness of the conceptual change. Considering the specific question that each of six of the students had originally answered *non-scientifically* on the computer diagnostic test and *scientifically* on the computer post-test, the six students each answered the same question scientifically on the retention test. The seventh student, who had originally answered two more questions *scientifically* on the computer post-test than on the computer diagnostic test, answered one of the two questions *scientifically* on the retention test.

Educational Importance of Study

These computer simulations could serve as a prototype remediation system for similar microcomputer simulations that would deal with many of the well-characterized alternative conceptions known to occur in the learning of physical science in grade eight (or nine). Such a remediation method would not require the input of considerable additional time and effort by the science teacher and would not require the purchase of specialized laboratory apparatus designed for the remediation of each different alternative conception.

Although the physical science student has not grasped completely the standards of what constitutes valid reasoning and sound evidence in the idealized system of science, computer simulations can help him or her to learn the canons of scientific rationality by *playing the game of science*. The student does this by encountering examples of situations that scientists have already learned to view as being different from other situations (e.g., ideal gas laws, free fall in a vacuum, and motion along frictionless surfaces). In this way he or she can build up a supportive web of familiarity with exemplars and nonexemplars of scientific thinking which will allow him or her to surmount the barrier of abstraction, judgment, and emotional acceptance between the naive concepts and the science concepts.

Bibliography

Dewey, J. (1916). *Democracy and education*. New York: The Free Press.

Champagne, A. B., Klopfer, L. E., and Anderson, J. H. (1980). Factors influencing the learning of classical mechanics. *American Journal of Physics*, 48, 1074-1079.

Gagne, R. (1962). The use of simulators. In R. Glazer (Ed.), *Training research and education*. Pittsburgh: University of Pittsburgh Press.

Garrison, J. W., & Bentley, M. L. (1990). Science education, conceptual change and breaking with everyday experience. *Studies in Philosophy and Education*, 10(1), 20-35.

Hasweh, M. (1988). Descriptive studies of students' conceptions in science. *Journal of Research in Science Teaching*, 25(2), 121-134.

Kuhn, T. S. (1970). *The structure of scientific revolutions*. Chicago: University of Chicago.

Kuhn, T. S. (1977). A function for thought experiments. In T. S. Kuhn (Ed.), *The essential tension* (pp. 241-265). Chicago: University of Chicago.

Lunetta, V. N., & Hofstein, A. (1981). Simulations in science education. *Science Education*, 65(3), 243-252.

Posner, G. J., Strike, K. A., Hewson, P. W., & Gertzog, W. A. (1982). Accommodation of a scientific conception: Toward a theory of conceptual change. *Science Education*, 66(2), 211-227.

Twelker, P. A. (1968). Simulations: An overview. *ERIC document*: ED25459.

Weller, H. G. (1988). Interactivity in microcomputer-based instruction: Its essential components and how it can be enhanced. *Educational Technology*, 28(2), 23-27.

Weller, H. G. (1990). *Microcomputer-based diagnosis and remediation of simple Aristotelian conceptions of force and motion*. Unpublished doctoral dissertation, Virginia Polytechnic Institute and State University, Blacksburg, Virginia.

Special Presentation—ISTE
Responsible Computing: Computers, Ethics and Students

Abstract

Although microcomputers have been in classrooms for at least a decade and we have witnessed marvelous gains in the power of computers and the quality of software, relatively few gains have been made in our understanding of the ethical use of computers and information. Increased access to information resources in electronic form has created a host of new ethical problems and has changes the context in which a number of ethical issues arise.

The panel addresses a number of concerns and issues in computer ethics from an educational and law enforcement perspective. It also suggests educational strategies that may be employed to help develop responsible computing in students. The panelists all presently serve as members of the Advisory Council of a Joint National Institute of Justice—U.S. Department of Education Computer Ethics project designed to help develop responsible use of computers by students.

Panelists:

Chairperson:
Jonathan Budd
Director, Computer Ethics Project
Program Manager, Law Enforcement, Computer Crime
National Institute of Justice
Panel Chair

Steve Purdy
Agent, U.S. Secret Service

Paul Resta
International Society for Technology and Education
Center for Technology and Education
University of New Mexico

Connie Stout
Education Program Director
Division of Technology Development
Texas Education Agency

Kathleen Hurley
Education Marketing
IBM Corporation

Sally Bowman
Director, Computer Learning Foundation
Palo Alto, California

Jim Mecklenburger
Director, Institute for the Transfer of Technology to Education
National School Board Association
Alexandria, Virginia

Special Presentation—ACM SIGCUE/ISTE

Technology Education for Teachers: Parts I & II—Focus, Activities, and Direction

Abstract

This session will introduce those in attendance to several special interest groups involved in technology education for teachers—both preservice and inservice. For each group, the focus, issues identified, activities, and direction will be briefly highlighted. The groups have evolved from different perspectives and membership backgrounds. It is important that the technology education community dealing with teacher education become cohesive and benefit from the activities and efforts of the others. Those in attendance will be involved in discussion. Unity of purpose and challenge will be explored with interaction of group representatives and attendees.

Panelists:

Henry Kepner, Jr.,
President, ISTE/SIGTE
Panel organizer and Moderator
University of Wisconsin
Milwaukee, Wisconsin

John Lawson
Chair, ACM/SIGCUE
Lewis Clark State College
Lewiston, Idaho

Jerry Willis
Coordinator, Annual Technology and Teacher Education Conference
East Carolina University
Greenville, North Carolina

Margaret Kelly
Chair, Association of State Technology-Using Educators
California State University
Fresno, California

Jackson Byars
Chair, Computer-Based Technology Education
Assoc. of Teacher Educators
Kansas State University
Manhattan, Kansas

Representative, American Association of Colleges for Teacher Education

Representative, AERA, SIG: Instructional Technology

Special Presentation-ISTE/SIGTC
The Role of Technology in the Restructuring of Schools

Abstract

This is a *Hypersession!* After brief opening remarks, the hyperpanelists, Alan November, Robert Pearlman, and Doris Ray, will serve as interactive, associative information bases for participants interested in exploring how technology can support and enhance restructuring schools. The participants will become part of the panel discussions, offering questions, comments, challenges, etc. The ensuing hyperlogue may also make use of videotapes, computerized images, or other audio-visual media.

Panelists:

Doris Ray
Director, Maine Computer Consortium
Auburn, Maine

Alan November
Glenbrook South High School
Glenbrook, Illinois

Robert Pearlman
AFT
Washington, D.C.

Projects
Hypermedia/Multimedia Applications

Animal Safari: Multimedia Learning for Special Needs Students

Christine Appert
Kluge Center
University of Virginia

Abstract

This presentation will provide an overview of Micro Mornings, an enrichment program for elementary-age children with learning disabilities and mild to moderate handicaps. Micro Mornings is sponsored by the University of Virginia Kluge Children's Rehabilitation Center and offers an intriguing sampling of organized small group activities during the summer and after school. Specifically, the Animal Safari project will be used to illustrate adapted computer-based experiences which take advantage of multimedia technology and integrated learning ventures.

The goals of the Micro Mornings program are to help students improve poor self-images, become more independent in learning tasks, take more responsibility for their own learning, and demonstrate the ability to cope with the learning process, including its inevitable times of frustration, confusion, and difficulty. Technology offers a vehicle for accomplishing these goals and provides a valuable medium for the flexible and individual planning required in special education. "Learning-centered" software, videodisc applications, and other interactive media provide a dynamic and meaningful context for learning which accomodates individual learning styles and abilities.

The Animal Safari project involved small groups, delineated by age, for six three hour sessions. Youngsters were introduced to various aspects of animal life through a variety of on and off-computer activities. Software selections includedpopular commercial programs,such as Sunburst's "Learn About Animals" or "Animal Trackers," Hartley's "Animal Survival," and National Geographic's "Project Zoo." In addition, *HyperCard* stacks were created to use with Videodiscovery's "Bio-Sci" videodisc and the "National Zoo" videodisc. Related projects encompassed a computer slide show with digitized images, creative writing, and charting. Sessions with the ASPCA or a localnature center complemented work with animal habitats and characteristics. Programs and activities were structured and adapted to adjust for variables such as deficits in reading and organizational skills.

Session participants will be given a description of the organization and special techniques employed in implementing this program. Specific suggestions for adapting software applications and creating program activities will be given. The unique benefits of multimedia for special populations will be highlighted throughout the presentation.

Hypermedia Instructional Enviroments: Learner Strategies and Learning Outcomes

Kim MacGregor
Louisiana State University, Baton Rouge

Abstract

Hypermedia, a rich instructional computing environment, offers possibilities for presenting information a format that allows for individual differences among learners. This project was undertaken to provide knowledge about secondary and college student's use of hypermedia. A hypermedia system was designed to present textual, graphical, and video information related to the subject area of interior design. A critical issue is whether students can navigate through cards and stacks purposefully without getting lost. Navigation through hypermedia systems requires that students demonstrate determines students' views of their control over the environment may be an important dimension of student success with this type of learning environment. formal measures of achievement and cognitive style provided information about student ability level and locus of control. Navigation through hypermedia may be influenced also by the use of instructional aids in the form of advance organizers. Two types of advance organizers (concept map or outline) were provided to students while they used the system. Information about the relationship between student characteristics, learner strategies, and learning outcomes will be presented.

White Plains Time Traveler—Local History on Interactive Videodisc

Kathy Rehfield-Pelles and Naomi Gross
White Plains Public Schools, New York

Abstract

During the Summer of 1990, four White Plains, NY educators developed an interactive videodisc computer project on the topic of Westchester history. Using resources and knowledge provided by fourth grade teachers who teach local history as part of their knowledge provided by fourth grade teachers who teach local history as part of there social studies curriculum, and with the assistance of Susan Swanson, Westchester County Historian and other local historians, the team compiled videos and text about the most important events and places in local history from the 1600's to 1940s. Over two hundred still or full-motion visuals were gathered from sites still standing or from photos and film clips in historical archives in the area. After puling together all the information, the team designed a Hyperstudio computer program for the Apple IIGS which allows users to view and interact with this history in non-linear ways. The events are indexed by time period and also by theme (Historical, Political, Economic, Geographic, Social) and can be accessed in any order. Users can follow events of a theme through different time periods or all of the events of a time through the different theme perspectives. This videodisc and computer program have become important resources for fourth grade teachers in there local history curriculum.

During the project presentation we will summarize the steps we took in designing and then producing our own videodisc and computer program. Emphasis will be place on ways that this project specifically takes advantage of the computer/videodisc medium (linking to next event to see, indexing swift and specific retrieval of visual data) and on the design elements we were careful to include that make this easy for the user to control and understand (mental mapping). We will demonstrate the completed project. We will discuss way in which fourth grade teachers are currently implementing the use of Time Traveler. We will briefly highlight other interactive video projects currently under development in White Plains.

Projects
Technology in the Elementary Setting

A New Direction: Distance Education and Chapter 1

Dianna Lawyer-Brook
RMC Research

Abstract

The Region 5 Rural Technical Assistance Center is currently piloting a distance education model using interactive television via fiber optic cable to deliver Chapter 1 instructional services. This experimental program is designed to provide math instruction to Chapter 1 students through the use of distance education technology at two locations, one in southeast Kansas and another in the panhandle of Oklahoma. Each location has a central sending site and three receiving classrooms. Students at a nearby school receiving the same instruction in a traditional classroom are used as a control group. The school districts provide the hardware equipment, while R-TAC is responsible for training, evaluating, and providing the curriculum.

The curriculum and methodology are aligned with current research, emphasizing the math process, problem solving, and the use of manipulatives/computers as mathematical tools. Teleteachers instruct the students five times per week on-line with paraproffessionals assisting in classroom management and homework projects.

Evaluation of this pilot focuses on outcomes form both the individual students and the program as a whole. Both formative and summative evaluation designs will be used to determine effectiveness. The results of this experimental program will prove useful to school districts as they make important decisions with respect to the use of technology in their schools.

Interpreting Literature Through Art and Technology: A Collaborative Project

Barbara Linsley
Mentor Teacher & Computer Trainer
Ontario-Montclair School District

Joan Hanor
Assistant Professor, Department of Education
University of LaVerne
Ontario, CA

Abstract

This project focuses on cross cultural interpretations of literature between Australian and Californian communities. It was implemented by two teachers, representing university and elementary levels, working together to establish creative channels for educational growth. Their purpose was to document ways technology could serve as a vehicle for examining, interpreting, and restructuring literature through visual analysis.

This pluralistic approach afforded students the opportunity to identify key elements of American Tall Tales, to express their reactions, ideas and feelings visually through art, and to communicate their analysis of related themes in collaborative groups using contemporary technological tools: AppleIIGS computers, video laser disks, Hyperstudio and 816 Paint.

Participation in WCCE '90 led to visitations to Australian Schools. This enabled first hand contacts in Australia for establishing the cultural exchange. Students there had shared their rich heritage of tales and an audio tape was provided for our students' benefit.

Current status of the project will be discussed along with goals, examples of learning activities, and samples of student work.

The WHAT'S IN THE NEWS Telecomputing Project

Chris Clark
Telecomputing Project Coordinator, WHAT'S IN THE NEWS
University Park, Pennsylvania

Abstract

Teachers and students in grades 4-6 dial the Learning Link network and receive up-to-date resources for a current events television series aired on public television stations. The goal is to enhance the effectiveness of educational television through telecomputing. Development began July, 1990, piloting in September. The service opened to Pennsylvania in January, 1991, and then nationally in March.

Telecomputing involves different levels of interaction. At one level, viewers receive program outlines, newspaper articles, library reference lists, thought questions, and schedule of programs. At another level, viewers communicate with program staff, students electronically submit news stories, and classes participate in monthly surveys. At the highest level, classes communicate with each other to conduct debates and carry out projects.

Celebrating its 25th season, WHAT'S IN THE NEWS is a nationally-distributed series. Its goals are to 1) put events and issues trends into perspective, 2) make news part of a student's life, and 3) strengthen current events education. Weekly 15-minute programs alternate between a summary of news events and a feature report on a single issue. Students become involved through essay-writing activities and "News Detective" segments. Thirty-one programs are produced annually by WPSX-TV in association with Penn State's College of Education.

Projects
Writing Across the Grade Levels

Implementing the Houston Independent School District Writing Labs

Dr. Richard Alan Smith
Director of Computer Curriculum
Houston Independent School District, Texas

Abstract

One of the most exciting developments in the Houston Independent School District is the recent establishment in each of our middle schools of computer laboratories dedicated to facilitating writing instruction. A gift of 1000 Deskpro computers by the Compaq Computer Foundation made possible the 38 labs, each with an average of 27 computers.

Our labs are unique because the students are taught to use the *writing process* to transform their thoughts into writing through six steps. The process begins with prewriting activities, writing a draft, and then discussing the draft; the students revise, edit, and finally evaluate the completed piece. The teachers were shown how to teach the writing process as they learned how to use the computers. Hence, rather than being simply an add-on to traditional instruction, the computers were seamlessly woven into a new state-of-the-art writing curriculum.

Accordingly, a visitor to any of the laboratories steps into an exciting environment where students are engaged in several activities. Teachers help students gather their ideas together, get those ideas down on paper, obtain suggestions for improvement from their fellow students, revise their papers on the basis of that feedback, edit their papers, and then evaluate their efforts. These activities are the key steps of the writing process. A frequent enhancement is publishing the papers in anthologies or newsletters.

Developing Creativity in Students through Multimedia

Stephen F. Bombara
Windham Public Schools, Connecticut

Abstract

Students are allowed to form groups of two or three, and given the task to design, manufacture, and market a real product.

Students will make their product known to the entire class to avoid any product duplications. They will have to manufacture a sample, and create a package for this product.

Students will name their company, develop a slogan, and draw a logo to be used for their letterhead. Students will learn to use the scanner to scan their drawn images.

During one class students will view favorite TV commercials and discuss the elements present, which make a commercial successful.

Students will learn how to use a word processor and desktop publishing package by drafting:

- A brief history of their company (how and why they got started).
- Give detailed information about their product (eg. recipe, ingredients, instructions).
- Develop a letterhead for their company.

Students will learn about databases and develop one by researching companies, that might buy their product, from a phone book.

Students will write to these companies asking for a time and place to meet for a sales presentation.

Prior to the meeting students will create marketing materials. These materials will include: transparencies of charts and graphs, created on a spreadsheet .

Students will write and act out a script for a radio commercial which will be aired over the school intercom.

The final task is to draft a script, make props, and design a set for the taping of a video commercial. These commercials will be shown at school based functions.

An Interactive Writing Program for Urban Teenagers

Randall Ryder
Milwaukie, Wisconsin

Abstract

Abstract not available at press time.

Special Presentation

Information Literacy: A Changing Education Environment

Abstract

Information is exploding using imagery, sound, and movement to influence and create impact. Text literacy is only one of many mediums for communication. Print, video, graphics, and sound each take very different approaches to reaching the mind. The impact of different information modes on thinking and learning will be demonstrated. Information presented will be research and experience based. Students can move from passive text information learning to knowledge building with dynamic interactive tools. Are your students going to be literate beyond print for reading and writing information?

Panelists:

Bernajean Porter
Senior Consultant
The Confluence International Group
Boulder, Colorado

Ron Fortunato
Senior Consultant
The Confluence International Group
Virginia Beach, Virginia

Special Presentation—ACM/SIGCSE

Computer Science: A College Preparatory Course for All High School Students

Abstract

In 1983 a task force working under the ACM Executive Board's Elementary and Secondary Schools Subcommittee drafted recommendations for courses in Computer Science at the secondary school level. These recommendations were published in 1985, jointly approved by ACM's Education Board and the IEEE Computer Society's Education Committee. Many changes have taken place since that committee completed its work. Among them are the report of the ACM Task Force on the Core of Computer Science and a curriculum model for an introductory course. The ACM Pre-College Committee have organized the task of developing a corresponding secondary school course. The proposed course is a one year, college preparatory upper division high school course for all students, comparable in scope and rigor to general biology, chemistry, or physics. Panelists will propose course models and solicit discussion.

Panelists:

Susan Merritt, Moderator
Chairperson, ACM Pre-College Committee and Dean
School of Computer Science and Information Systems
Pace University
Pleasantville, NY

Philip East
Associate Professor and Acting Head, Department of Mathematics and Computer Science
University of Northern Iowa

Darlene Grantham
Secondary Teacher Specialist
Montgomery County Public Schools
Maryland

Charles Rice
Mathematics Department and Computer Coordinator
The Dalton School
N.Y., N.Y.

Carol Wolf
Chairperson, Department of Computer Science
Pace University
N.Y., N.Y.

Lesley Klein
IS Department
Panel Coordinator
Pace University
Pleasantville, N.Y.

Projects
Using Computers to Collect, Analyze, and Synthesize

Hands on Data: Putting Kids in Charge of Data!

Phyllis Kalowaki
Newton, MA

Abstract

The goal of the Hands On Data project, supported jointly by TERC and the Cambridge School Department, was to develop interdisciplinary units in which students formulate a problem, gather and interpret data, and communicate their conclusions. This process was aided by use of prototype software, the Tabletop, which helped translate NCTM standard regarding data analysis into lively classroom activities.

This session will provide a practical review of a year long research project involving a fifth/sixth grade and an eighth grade class. By peeking into the classrooms, the results of using technology along with inquiry learning activities will be discussed and the impact examined. We will analyze how teachers, In both a self-contained and departmentalized situation, went about developing these units. Some issues arose regarding curricular integration, integration of technology, classroom management, and teacher support. The efforts and progress in resolution of these issues will be discussed.

The Tabletop: A Visual Tool for Creating, Exploring, and Analyzing Data

Chris Hancock
Project Director, Hands on Data
Technical Education Research Centers, Inc. (TERC)

Abstract

The Hands On Data project has been developing and piloting a data exploration and analysis tool called *The Tabletop*, as an exemplar of a new class of tools for working with data. The Tabletop is a highly visual, highly interactive interface that responds to queries visually through animated icons on the screen. the software has the ability to arrange these icons into Venn diagrams, graphs and scatter plots according to attributes determined by the user. This visual representation allows concepts ordinarily considered "advanced" to be understood at a surprisingly early age. The software's quick feedback encourages students to explore and to experiment with their own ideas.

This presentation will demonstrate the Tabletop, mention some principles underlying its design, and discuss its intelligibility to students of various ages. It will also consider briefly what kinds of skills and concepts underly the use of computerized data tools and the new kinds of mathematics, and new meaning of the term "computer literacy," which these skills and concepts suggest.

The Hands On Data project is conducted by TERC in collaboration with southeastern Massachusetts University. It is funded by the National Science Foundation with additional support from Apple Computer, Inc.

Exploratory Data Analysis, Median Polish, and Mathematica

Martin Adickes and Rob Rucker
Industrial and Management Systems Engineering and Center for Computer Integrated Manufacturing
Arizona State University, Phoenix

Abstract

The work presented here grows out a need to teach undergraduate engineering students approaches to analyzing real world data. These approaches are designed to help them in their initial investigations of data sets, with a view toward detecting unusual or important features prior to formal hypothesis testing. Data from real environments does not usually lend itself to easy application of standard confirmatory statistical analysis techniques and so other, complementary, approaches are often necessary. One such approach is that represented by the field of Exploratory Data Analysis (EDA). Within that field we have chosen a particular technique, Median Polish, to introduce to our students. The first part of this paper explains median polish and associated techniques, in an introductory manner. The last part of the paper describes the mathematical environment available within the computer package *Mathematica*. An appendix presents the outline of our interactive tutorial on this topic. We have written the tutorial in a 'notebook' style using the programming tools within *Mathematica*, an environment for doing mathematics. This computer environment, together with accompanying text material, allows the student to experiment with the concepts and the consequences of analyzing data in an exploratory manner. The material has been presented to students who had taken an introductory statistics course and were 'Macintosh literate.' The results were positive on the whole but pointed out the need for supplementary instruction in more basic skills such as finding medians and matrix operations in general.

Projects
"Take a Risk" Students with Special Needs

Enhancing Learning through the Use of Interactive Video

Doris Tyler
Chapel Hill High School, North Carolina

Abstract

The Sophomore Seminar: Program for Academic Challenge currently involves 85 students and is designed to enhance student's ability tostructure, analyze and communicate information across the content areas of social sciences, science, mathematics, and English. Through an interdisciplinary, project-oriented curriculum, we are helping students to see themselves as successful learnersand capable young adults. A four-hour block of uninterrupted time facilitates a true interdisciplinary approach, helping students understand the connection of these disciplines with the real world.

Two major focuses of all seminar activities are to improve student self-esteem and to enhance skills related to accessing, processing and communicating information. Students were placed in cooperative learning groups to research and develop a multimedia presentation for biology. Traditional research skills as well as on-line databases were used to locate information in the media center.

Students were taught to use the Macintosh computer, *HyperCard*, and a videodisc player. Student created interactive video stacks were created using the Life Science videodisc from Optical Data. Each research presentation included information about one phylum or class of organisms; the information presented depended upon the research as well as the group of organisms but included the general body structure, the geographic habitat, place in the food web, and usefulness to humankind.

Mac-in-Touch

Diane Frank, Carol Goltz, Mary Margaret Stephens
West Anchorage High School, Alaska

Abstract

The three presenters are completing the second year of an Apple "Equal Time" Grant which supports an interdisciplinary project targeting at-risk ninth graders and which emphasizes the development of *HyperCard* stacks and interactive media. Enthusiasm for *HyperCard* has put the Mac in touch with kids and curriculum throughout the school. This presentation will show the evolution of interdisciplinary HyperCard stacks developed by students ranging from at-risk freshmen to advanced placement senior, demonstrating stack development from the simple to the sophisticated as students become more comfortable and creative with the technology.

From Desktop Publishing to Hypermedia

Kathleen Sutphen
T.E. Mathews County Community School
Marysville, CA

Abstract

T.E. Mathews County Community School, located in Marysville, California, was established to serve at-risk youth. Our 45 students range from grades 6 through 12. The majority of our students have low academic skills, low self-esteem, and little or no history of success within a school environment.

Against this setting we were awarded and "Equal Time Grant" by the Apple Computer Corporation for the two consecutive school years, 1989 through 1991. The focus of our grant was to have our students publish a community newsletter, *The Other Side*, using a fully networked Macintosh computer lab donated by Apple. Our major software program for publishing is Aldus *PageMaker*. Our first year was an unheralded success.

During the 1990-91 school year we combined a California Technology Grant with our Apple Equal Time Grant. The focus of our new grant was to augment our existing technology use with use of hypermedia. With some new equipment and software, our students began to create a *HyperCard* based software program that gives the potential user an interactive medium to learn about the history of our culturally rich geographical area. We produced our newsletter in conjunction with development of our software program, using researched topics, interviews, pictures, recordings, etc., for both products.

We have come a long way over the past two school years. The evolution of technology has been fast-paced and we and our students have run a successful footrace keeping up with it. We are all enthusiastic. Our presentation will highlight what we've learned, "tricks" that helped make it work, and a sampling of our students' hard work.

Projects
Teacher Education Telecommunications

Using PROLOG and Hypertext in Undergraduate Teacher Preparation

Gary Tubb
Louisiana Tech University

Abstract

The presentation will focus upon documented uses of PROLOG and hypertext in an undergraduate course that integrates computer education with media. Techniques for teaching LOGO in the course are contrasted with teaching PROLOG. Emphasis is placed upon graphics and database notions that each language conveys. Experience with the construction of a simple expert system for media selection, as part of the course, will be shared in the presentation. Finally, experience in using a simple text-based hypertext system in the course will be shared in the presentation.

Education Students On-Line at Cal State Los Angeles

James H. Wiebe
California State University, Los Angeles

Abstract

This presentation will describe how the Instructional Technology program at California State University, Los Angeles began its own BBS; how the BBS is maintained and run; how it is used in introductory courses for teachers and in our graduate programs; and our plans for it.

Our BBS was started two years ago on an IBM-XT clone with a 20 megabyte hard drive and using a public domain BBS program, *Opus*, Version 1.03. We are a node on FIDONet, a national network which allows our students and faculty to send electronic mail, access various forums, obtain public-domain software, etc. Our BBS has been enhanced through several small grants, including one which allowed us to purchase 15 modems to lend to students so that they could use the system from home and another which allowed us to develop a variety of curriculum materials to use in introductory computer courses for teachers. We recently logged our 5000th call. Fifty-five percent of these calls have been course-related.

Our enhancements to the system have dramatically increased usage of and user satisfaction with the system. A recent Instructional Technology MA project evaluated usage of the system. It was found that there has been steady growth in use and that the system has enhanced faculty-student interaction outside of class time.

Wednesday

NECC '91 CONFERENCE PROCEEDINGS

Special Presentation

Education on the Electronic Frontier: Learning and Teaching on Tomorrow's Networks

Developments in computing and telecommunications have made possible new forms of learning and teaching. This panel will address a number of questions raised by these developments. It is now possible for students and teachers to interact with adults outside the school system. What educational models will support this kind of interaction?

With access to expertise and information outside the classroom, teachers and students can become true inquirers. What changes will be necessary for such inquiry to take place on a large scale in education? What sources of expertise and information should teachers and students be able to access? How can students and teachers be taught how to locate what they need when they need it? How do kids and teachers learn to make sense of all the information they get, once they tap into it? What incentives are there for experts to provide services to kids and teachers "on demand"? This panel will address these questions from several different perspectives, including those of K-12 schools, higher education, government and business.

Panelists:

James Levin
University of Illinois

Michael Waugh
University of Illinois

Beverly Hunter
National Science Foundation

Margaret Riel
AT&T Learning Network

Al Rogers
FrEdMail Foundation

Projects
Project Outcomes

Computers and Classroom Instructional Research

Shirley Becnel
Jefferson Parish Public Schools, Louisiana

Abstract

For the past three years, the Chapter 2 department of the JPPSS has funded more than 20 research based instructional projects for the purpose of fieldtesting and evaluating innovative programs in the classrooms prior to recommending district wide changes in the instructional program. Six of the completed, successful, projects used computer technology as an essential part of the instructional design. As a result of the fieldtests, three of the projects (1,2,3) have been expanded and incorporated into district curriculum. Three other projects (4.5.6) yielded data which were used in the restructuring of specific subject area guides.

The purpose of this presentation is to report the results of six technology based research projects. These projects are: 1) a district wide high school writing project which addresses both the general writing/editing skills of secondary students and the specific skills needed to pass the writing portion of the state high school competency test; 2) a district wide science technology project using computer probes and sensors in biology, earth science, and physical science classes; 3) a high school biology project using videomicroscopy; 4) two computer based writing projects, one involving middle school special education students and one for handicapped primary level special education students; 5) an algebraic graphing project utilizing large screen computer display units and graphing software in pre-algebra classes at the middle school level; and 6) a computer based reading vocabulary project for over-aged, academically at-risk, non-special education elementary students in grades 5/6. Note: Project 1 was implemented at 7 high schools and project 2 involved students in 8 high schools and 17 middle/junior high schools. Other projects involved one or more classes at a single school site.

Participants will receive a description of each project including computer and equipment specifications, copies of the project designs, evaluation data, and the final reports.

Technology Now and Tomorrow—An Interactive Approach to Elementary Science

Beth Sandez
North Ridge Magnet School
Moreno Valley, California

Abstract

In a time when the drop-out rate is rising and scores on standardized tests are declining, the question for educators is: How can we equip students to be independent lifelong learners capable of creating and utilizing tools to solve problems and improve life? Since today's elementary school aged children were born into the world of Nintendo, VCR's, and electronic games as toys, the answer seems to lie in the area of interactive technology. Paper and pencil are simply not sufficient tools for today's technology aware students.

Schools are gradually learning from business and industry to use technology as tools to instruct and motivate students. One of the most exciting uses of technology in the elementary school is interactive laser disc programs, especially in the area of SCIENCE. There are wonderful laser disc programs which bring the world of space, nature, or earth science right inside the classroom! Come to see what a Science, Math, Technology Magnet elementary school is doing to prepare its students for the twenty-first century!

Indiana's Project 3R's

Julie Poage
Ball State University, Indiana

Janelle Zorich
Apple Computer Inc.,
Indianapolis, Indiana

Abstract

On June 13, 1990 Indiana's Governor Evan Bayh and the Superintendent of Public Instruction, H. Dean Evans, announced the implementation of **Project 3R's**, a program designed to extend educational technologies in elementary schools throughout Indiana. The program made available $2,000,000 to aid in the establishment of educational technology laboratories for instruction in elementary reading, writing, and mathematics, focusing primarily on kindergarten and/or first grade programs.

The project entailed the completion of three components: (1) petition and slection of participating school corporations; (2) the training and support, including hardware, software, and curriculum development; and (3) evaluation of the project through the maintanence of student achievement indicators.

The Indiana Department of Education evaluated school corporation petitions on the basis of how well they addressed the intent of the enabling legislation (HB 1290, Chapter 25, Sections 1-4) and the extent to which the implementaion plans addressed specific guidelines, such as the commitment of matching funds and staff support.

The training and technical support associated with the use of Apple equipment included regional technical support, training sessions with *Apple Learning Series: Early Language Training*, and curricular integration specific to language arts, writing, math, and problem solving provided by staff at Ball State University. Follow-up training sessions have been scheduled at regular intervals throughout the 1990-91 academic year and will address specific needs expressed by participating teachers.

Evaluation of **Project 3R's** will include pre and post survey data concerning identified needs and staff development expectations of project participants, as well as descriptive statistics of student achievement gains. Qualitative data will also be collected from randomly sampled school corporations, including the level of impact and satisfaction with **Project 3R's**.

Special Presentation

Visual Arts —A New Frontier in Computer Technology

Nancy Scali
Ontario-Montclair School District
Ontario, California

Visual Arts features an Arts Education Technology presentation developed for the J. Paul Getty Trust for Education in the Arts. Emphasis will be placed on the creative and technological processes used in the development of a child or adult's computer artwork. The communication, artistic expression and pictorial interpretation of creative visual thinking will also be explored. A mini gallery of professional and student computer artwork will be on display. Discipline Based Art, Fine Art and curriculum integration via Art will be shared.

Fine Arts projects featured in *The Writing Notebook* and Sunburst's *Solutions* will be on display.

Projects
Computers in the Future

Imperative: Technology Training for Administrators

Hilary Cowan
Kanawha County Schools, West Virginia

Abstract

The Effective Schools research emphasizes the crucial importance of strong and purposeful leadership by the principal. At the same time, advocates of restructuring education have emphasized the need for effective use of technology to increase student productivity. Despite the general agreement on these issues by national educators, almost no attention has been given to the training that will be needed by school principals and central office administrators to understand their role in achieving technological excellence in their schools.

If we are to assume that administrators need training to become effective technological leaders, then we must address the question of how to achieve that goal. This includes an understanding of what components need to be developed in order to teach administrators what they need to know. It also needs to include a reality assessment ot the time and effort that can be expected from administrators who are already under pressure to implement other major insructional strategies. Furthermore, one cannot ignore the all too common fear by administrators that their ignorance of technology will be exposed and will impact their role as a leader.

For these reasons, training must be developed around three concepts: 1. limiting the time and number of sessions while maximizing learning, 2. limiting the training classes to peers, so that administrators can feel free to make mistakes and ask important questions, and 3. focusing the training on effective`*supervision* of technology rather than on effective *use* of technology.

Although aspects of the training will be identical with that which is developed for teachers, certain issues are unique and others must be expanded or reduced to fit the administrative needs. It is important, for example, that principals understand their unique role in establishing a consistency in the school use of technology. It is also essential that they learn how to evaluate the use of technology by teachers in classrooms. On the other hand, although some hands-on time with software is necessary, administrators do do not necessarily need to learn how to use any program in great detail. It is far more important for them to be able to assess whether the program itslef is consistent with county, school, and classroom instructional goals.

The above issues are a few of those which will be covered in more depth in the presentation. Practical considerations, such as the politics of maximizing participation in training will also be discussed.

If technology is to move forward in schools across the nation, then we no longer have the option of simply training our teachers andand hoping that consistency of purpose and excellence of technological use will filter through our classrooms. The time for emphasizing the need for administrative training is now.

The Florida Science Videodisc Project

Robert Fronk

This is an update on the 1991 Florida Science Videodisc Project. The initial 1990 project has received further funding and is being continued into 1991. This year 30 Florida elementary and middle/junior high school teachers will get together at Florida Institute of Technology in Melbourne to learn about and work with computers and videodiscs. In addition to receiving training in interactive video development in the project's educational technology laboratories, participants will receive a videodisc player, a stipend, room and board for those staying on campus, videodiscs, computer software, graduate credit, and other materials. The project will create the second "Florida Science Videodisc" with slides and video from the 30 teachers and F.I.T. faculty. The 1991 project will focus on the life sciences (last year's work dealt mainly with the physical sciences). The 1991 Florida Science Videodisc will include all of the nearly 2000 physical science images from last year's project in addition to approximately 2000 new life science slides from this year's project. As long as the supply lasts, the 1990 Florida Science Videodisc (with its associated software for Apple II, Macintosh, and MS-DOS) may be purchased for $20.

The School of the Future: Visions for Learning

Gwen Solomon
The School of the Future, New York

Abstract

The future is now! New York City's School of the Future, a public secondary school, opened in September, 1990 in midtown Manhattan. This is our vision of education for the future: the methods and technologies we use to help our students learn 21st century skills.

We hope to help students get, understand, and manipulate information, learn how to learn, think creatively, make decisions, solve problems, and work collaboratively.

Our students are engaged in active learning; they create presentations and performances. They collaborate on projects to pool strengths and learn from one another. We have creative scheduling so students spend longer periods of time working in depth at their tasks. We have an interdisciplinary Humanities (English and social studies) curriculum that is based on a thematic approach to American history. We find that technology provides wonderful and important tools for student learning.

Many elements help us achieve success as we create a learning environment for the 21st century.

Projects
Mathematics Applications: K-University

Ease Students into programming with Turbo Hal

John T. Taylor
Hillsborough Community College
Tampa, FL

Abstract

TURBO HAL: the Integrated Environment, has been developed as a tool introducing students to structured programming, especially in beginning Pascal classes. It is designed to be a low cost alternative programming simulator to Karel, the Robot—a well-known such simulator. It is an easy-to-use integrated environment that does not require a seperate editor, compiler, linker, and debugger in order to create, debug and run Karel programs. HAL lives in the Borland language environment with its editor, windows and pull-down menus. It thus eases the student into a popular programming environment. This software was designed by Tim Totten of Hillsborough Community College

Turbo Hal has been successfully used with elementary and junior high students. Initially, a lifesize three dimensional grid made from a bed sheet and felt strips is used as a "playing board". The students are given programming template handouts on which they write-in the primitive commands to solve a typical HAL problem in the 3-D world. A student then walks through the grid as though he/she was HAL following the student's program. While observing "HAL" walking through the grid, the other students eagerly check and "debug" their own programs to solve the problem.

Since HAL can only make a limited number of simple moves. constructing more complicated moves, the students must use, and thus learn, the concept of a procedure. Older elementary and junior high students are easily introduced to iteration and conditionals as well as loop structures. By using a combination of the 3-D world and HAL's 2-D world on the computer, students are successful programming solutions to more complex mazes.

Teachers as well as students may generate their own problems by using the HAL'S WORLD generator or watch HAL execute step-by-step instructions solving many different HAL problems

I have designed a hypertext environment for HAL, using IBM's LInkway. It includes sample tmeplates for the beginners and instructions on how-to-build HAL grids. From any "page" (screen), the user may access a lesson or slide show to review or reinforce the programming concepts. Turbo Hal may also be accessed at anytime from any screen.

Turbo Hal runs on low-end MS-DOS machines and operates in text mode so that no graphics card is needed. It is available on a 5.25" disk. the Linkway version requires a mouse, CGA graphics and 640K RAM. It is available on a 3.5" DD disk. All software is distributed by Hillsborough community College.

All participants will receive copies of Turbo Hal for their school's use. a runtime version of the linkway hypertext presentation is available for the price of an exchange disk.

Using Hypertalk Scripting to Introduce Beginning Algebra Concepts

Barbara Dubitsky
Bank Street College

Rita Kabasakalian
New York City Lab for Gifted Education

Abstract

We have found that children learn mathematics more easily and with deeper understanding when they are engaged in solving mathematical problems that interest them. Certain kinds of computer programming activities provide interesting problems to enrich the mathematics classroom. We had been using the Logo language to provide these experiences, but this year we began using Hypertalk.

Seventh and eighth grade students in The Lab School, a public junior high school in New York City were taught Hypertalk scripting as part of their regular mathematics classes. The varied uses of variables as well as the use of graphics with a coordinate system were emphasized.

The concept of the variable was easily understood as the students explored the commands "ask" and "answer" and the uses of local variables with both words and numbers. Some children needed the "repeat" and "repeat with" commands for their work; several of these children easily understood recursion, others could use it only mechanically. We introduced graphics and animation possibilities within scripts, these required extensive use of a coordinate system. We found the rotation possibilities to be of particular interest to the students.

When we then used these concepts in more traditional mathematics work, we and the students referred often to the Hypertalk experiences. We found that all the students could easily switch from the Hypertalk coordinate system to the usual Cartesian system and that they understood the use of variables in their algebra work.

MathTech
A Collaborative Project of PLAYING TO WIN and COMMUNITY SCHOOL DISTRICT #4 (NYC)

Andrea Kimmich-Keyser (Playing to Win)
Gayle Rubenstein (District #4, New York City)

Abstract

MATHTECH was planned as a collaborative project involving parents, children, and their teachers in exploring technology-using extensions of the children's math curriculum. Teachers from fifteen elementary and intermediate schools, each with a mathlab equipped with an Apple IIe computer, participated. students and their parents attended weekly out-of-school computer sessions at the Playing to Win Computer Center.

In a typical MATHTECH project, and following in-class use of Math Shop (Scholastic software), students and their parents surveyed local stores, interviewed proprietors, collecting examples of real life use of mathematics, and developed a neighborhood directory of stores along with a descriptive map showing their locations. In the process, both children and parents learned to use word processing to write up the interviews, dat bases and spreadsheets to record survey results, and computer graphics to create the map.

This activity, along with other MATHTECH activities, will be described in a manual for teachers in other communities who wish to promote parent child collaborative math learning.

The presentation will describe some of these other project activities, show samples of student work, and report on results obtained.

Paper

K-12 Computer Teachers Literacy Status and Training Assessment in Illinois

Luann Stemler
Joaquin Vila
Illinois State University

Abstract

The purpose of this study is to report the current literacy status and training needs of K-12 computer science teachers in the State of Illinois. In order to gather comprehensive K-12 information on computer teachers, a survey was sent to those individuals whose major teaching assignment was computer related in public school systems. The total population of such group was surveyed (Illinois State Board of Education Data Base). Study findings, based on a 54% response rate, included the following: (1) Only 9.4% of the respondents hold a degree in computer science; (2) 85% have taught BASIC programming at some time in their career, but only 6.5% of those teachers have formally studied the language; (3) Over 95% said they are using microcomputers in the classroom as opposed to minicomputers or mainframes; (4) More than half of the K-12 computer teachers have interest in attending graduate courses or workshops to learn about multimedia technology in the classroom.

Introduction

The computer is now a familiar sight in most schools. Taylor and Poirot (1984) stated that the movement of computers into the school system has presented a problem in the lack of teachers trained in the use of computers as an aid to the educational process and a lack of teachers who can teach about computers. While computer utilization continues to expand in the schools, the training of teachers to make use of the computer and to teach the topic of computing has significantly lagged behind.

Poirot, Taylor and Norris (1988) survey the state computer educator coordinators and found that most states are able to find faculty to teach high school computer science. The coordinators expressed doubt about finding quality faculty to offer the more advanced courses anticipated in the future, particularly when computer science certification standards are enforced. The conclusion was drawn that retraining is critical if computer science is to develop as a discipline in the high school.

The ACM Task Force on the Core of Computer Science anticipates and demands high school courses provide a computer science background far beyond the elementary level courses commonly found in the high school today. Cooperation between universities, state education coordinators and districts are needed to develop and implement programs that provide for the special needs of the local teachers. The computer teacher needs to become a true computing professional who can adapt to a changing, expanding curriculum.

The study was undertaken in order to better understand the existing status and training needs of the K-12 computer science teacher relative to the teaching of computer science in the public school systems of Illinois. The survey also grew from a desire in the Applied Computer Department at Illinois State University to develop variety of courses relevant for the K-12 computer science educator. Casual observation had led to the conclusion that the present quality of instruction covered a wide range, with some instructors doing a fine job in the classroom, while there are some instructors who sorely need to upgrade their computer skills.

Research Methodology

Population

The survey was mailed to K-12 computer teachers in public school systems in the state of Illinois. The population source came from the Teacher Service Record collected by the Research and Evaluation Department of the Illinois State Board of Education. All individuals whose major teaching assignments was categorized as data processing, computer literacy/awareness, or computer programming were surveyed. Surveys were sent to 627 computer educators and received 340 responses, a 54% response ratio.

Survey Instrument

A survey questionnaire was constructed that consisted of items assessing four areas of interest: (a) teacher background, (b) computer literacy assessment, (c) computer hardware (d) training interest.

- The teacher background component surveyed major area of concentration and degree held (BS/BA, MS/MA and PhD.)
- The computer literacy component assessed credit computer science hours taken. Those hours were further analyzed by programming languages, Data bases, literacy courses, etc.

- The computer hardware component inquired about existing computer resources available at their school.
- The training interest component explored the current and future areas of interest of the computer science teacher. An extensive list of possible courses were presented including: multimedia technology, Computer Aided Instruction, Teaching Strategies for programming instruction, etc.

Findings

The respondents in this study reported only 9.4% of the computer science teachers in the State of Illinois hold a degree in computer science. Business, Mathematics, and Elementary Education are the predominant disciplines. (See Table I)

Only 9.1% of the responding computer teachers have taken less than 5 hours of credited computer science courses. Whereas, 47.9% have taken over 16 hours. (See Table II)

Table III displays the credited courses taken by the target population. 74.7% of the respondents took Introduction to Computers and 77.4% have taken some programming course.

Table IV shows the programming languages that are being taught in the K-12 school system, and further shows the interest areas of the classroom teacher. 85% of the computer teachers have taught BASIC programming at some time in their career. Surprisingly however, only 6.5% of those teachers have formally studied the language. Furthermore, the questionnaire indicates that 30.3% of the respondents are interested in learning the C programming language followed by Pascal.

The survey revealed there are several ways the teachers are coping with the ever changing field of computer science. 36% of the respondents have attended graduate courses. 64% have attended computer related workshops. Table V shows how computer teachers are learning on their own.

Table I
Major Areas of Concentration

	BS/BA	*MS/MA*	*PhD*	*Total*
Accounting	4.1%	0.0%	0.0%	4.1%
Business	24.1%	15.6%	0.6%	40.3%
Computer Science	3.5%	5.3%	0.6%	9.4%
Economics	0.9%	0.6%	0.0%	1.5%
English	2.9%	0.9%	0.3%	4.1%
Elementary Education	13.2%	9.1%	0.0%	22.4%
Foreign Languages	0.6%	0.3%	0.0%	0.9%
History	1.5%	0.3%	0.0%	1.8%
Home Economics	0.3%	0.3%	0.0%	0.6%
Industrial Technology	0.3%	0.9%	0.3%	1.5%
Mathematics	23.8%	11.2%	0.3%	35.3%
Science	5.3%	2.6%	0.0%	7.9%
Other	0.6%	7.9%	0.6%	9.1%

Table II
Computer Science Credited Hours

0 - 4 hours	9.1%
5 - 8 hours	9.4%
9 - 12 hours	17.4%
13 - 16 hours	12.4%
more than 16	47.9%

Table III
Credited Courses Taken

Introduction to Computers	74.7%
Word Processing	40.3%
Data Base	30.0%
Spreadsheets	29.1%
Desktop Publishing	9.4%
Microcomputers in the School	42.6%
Programming	77.4%
LOGO	21.1%

Table IV
Programming Language Course Status

	Courses Taught	*Courses Taken*	*Courses Interested in Learning*	*No Interest*
BASIC	85.0%	6.5%	2.4%	2.6%
Pascal	19.7%	23.2%	19.7%	16.5%
COBOL	21.2%	19.1%	10.6%	22.6%
Assembler	5.9%	20.6%	12.1%	27.1%
C	1.5%	2.1%	30.3%	24.1%
PL/I	1.2%	9.4%	10.6%	32.9%

Table V
Self-Instruction

Reading computer related articles	78.5%
Reviewing new software packages	75.6%
Any combination of the above "all by themselves"	67.9%

Most K-12 computer teachers responded that they are using the microcomputers (95.5%) as opposed of minicomputers or mainframes. Figure 1 depicts the ratio of microcomputer type used.

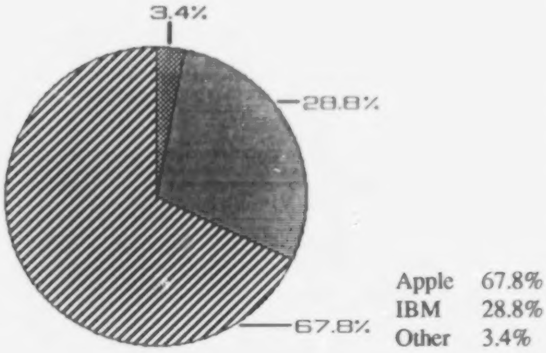


Figure 1. Microcomputer Type Used

According to the survey, there is broad interest in attending graduate courses and workshops. Table VI shows distinct areas of interest.

Table VI
Computer Area Interest

Introduction to computers	4.1%
Programming Courses	22.0%
Structured Programming	15.6%
Teaching strategies for programming instruction	39.1%
Hardware \ Computer Architect	22.4%
Software Design	25.0%
Using Multimedia Techniques in the classroom	51.2%
Artificial Intelligence	31.5%
Computer Aided Instruction:	37.1%
Authoring Software	22.4%
Gradebook	17.1%
Wordprocessing	22.0%
Spreadsheets	25.0%
Database	24.7%
Desktop Publishing	45.9%

Conclusion

Overall, the survey proved very fruitful. Information collected on the K-12 Computer Teacher Questionnaires satisfied the objectives formulated for the survey. The K-12 computer teachers current literacy status and training needs were assessed. The outcomes of the survey indicated that formal training is wanted to strengthen concepts taught at K-12 level as well as to facilitate the instructor mission through proper professional development.

Survey results were also useful in identifying and ranking the topical areas for (credit and non-credit) workshops and graduate courses. The training needs emphasized by computer teachers were in the areas of multimedia technology, strategies for programming instruction, computer-aided instruction, and artificial intelligence.

References

Denning, P.J., Commer, D.E., Gries, D., Mulder, M.C. Tucker, A., Turner, A.J., & Young, P.R. *Draft Report of the ACM Task Force on the Core of Computer Science*, February 1988.

Poirot, J.L., Taylor, H.G., & Norris, C.A. "Retraining Teachers to Teach High School Computer Science. *Communications of the ACM*, 31, pp. 912-917, 1988.

Taylor, H.G., & Poirot, J.L. "Computer Science Teacher Certification: Current Status and Trends." *T.H.E. Journal*, pp. 103-107, 1988.

Paper

Experiences Retraining Elementary and Middle School Teachers To Teach Computer Programming

Judith D. Wilson and Robert M. Aiken
Computer and Information Science Department

Glenn E. Snelbecker and Nina P. Bhote
Educational Psychology Department
Temple University

Abstract

A project implementing a model retraining program in computing and computer technology for elementary and middle school teachers is described and assessed. The project, still in process, is retraining teachers to become teachers of computing and computer resource people for their schools. As computing teachers, these teachers will be expected to teach computer programming, and programming is normally taught at the elementary and middle school levels with the objective of developing mathematical, problem solving, and learning skills. Although the project has been very successful as judged on several objective measures, informal observations suggest that the courses that teach programming may need to be modified to reflect changing views about the needs of teaching computing at the elementary and middle school levels.

1. Introduction.

Many educators would agree that elementary and middle school students can benefit from using computing tools to support knowledge acquisition and problem solving activities throughout the curriculum. Some envision a revised curriculum where computing plays an integral and interdisciplinary role (Burke, 1990; Goodspeed, 1989; Haney, 1990). Many would also agree that the ability to program a computer is an important skill to have in a computer-based information society. Additionally, although the supporting evidence appears inconclusive, many educators believe that learning to program a computer can have beneficial side effects for students at all grade levels, including reduced apprehension about mathematics and development of general learning and problem solving skills (Papert, 1980; Pea, 1989).

Educational software tools have been designed to facilitate learning in many subject areas in the K-9 curriculum, including mathematics, science, and language. Numerous software packages are now available and new packages enter the educational market frequently. To benefit from evolving computer technology, schools need teachers who are able to evaluate and select hardware and software to support educational objectives, integrate computer resources into the curriculum, and train teachers to use new technology. Teachers trained to conduct such activities are known as computer resource people or "teacher technologists".

We distinguish the teacher technologist from the computing teacher. Computers are the focus of courses in computing which may include introductions to computers and applications packages but which generally teach computer programming using either Logo or BASIC. In elementary and middle schools, programming instruction usually is aimed at fostering mathematical, problem solving, and learning skills. Thus elementary and middle schools need teachers prepared to function both as teachers of computing, teaching problem solving and learning skills through programming, and as teacher technologists.

Considerations such as these have influenced the design of a model program to retrain experienced elementary and middle school teachers to become teachers of computing as well as teacher technologists. We describe and appraise an implementation of the model which is funded through a National Science Foundation project grant. Now that the project is more than two thirds completed, it is possible to make a preliminary assessment of it as a vehicle for retraining teachers with little or no computing experience to teach computing. Although the project has been very successful generally in achieving its objectives, there are some early indications that those aspects of the model program concerned with retraining teachers to teach programming may need to be modified.

2. A Model Retraining Program for Elementary and Middle School Teachers.

The retraining program, which is still ongoing, is being conducted by Educational Psychology and Computer Science faculty at Temple University in cooperation with the Philadelphia School District. Teachers from the school district were selected to participate in the project on the basis of motivation and likelihood of success as indicated by past achievement. Previous computing experience was not used as a selection criterion, and care was taken to select participants from a variety of different backgrounds.

2.1 Curriculum Goals and Structure.

The goal of the model retraining program for elementary and middle school teachers is to prepare teachers to perform the broad array of functions expected of computing teachers in elementary and middle schools. Thus the general goal of the project may be viewed in terms of several component goals: (1) preparing teachers to become teacher technologists, (2) training teachers to teach computing, and (3) focusing on teaching strategies for computing courses that foster the development of general problem solving and critical thinking skills. Measures of success in the attainment of these goals include having more than 50 percent of the teachers complete the program, and having graduates of the program pass a comprehensive computing examination administered by the school district in order to be eligible to teach computing in the elementary schools (Aiken, 1988).

The model was designed to provide in-depth training in computing within two years while teachers continue their normal teaching duties. The program curriculum consists of five courses which have been designed to achieve all three project goals. The principal objective of the first and last courses is to prepare teachers to become computer resource people, while the middle three courses teach programming with a focus on how programming can be taught effectively to young children. Moreover, an underlying principle is that project participants would have a broader perspective about computer programming once they had seen what applications programs can do and how they function in the first project course.

The first course covers elementary computer concepts and introduces applications programs and their use in the classroom. The next two courses teach Logo programming at the beginning and advanced levels with an emphasis on modular and top-down programming concepts and on how problem solving using Logo can generalize to other types of problem solving tasks. The fourth course teaches problem solving with structured BASIC and prominantly features problem solving strategies, top-down design and structured programming concepts. The final course addresses educational software evaluation and effective uses of computers for teaching.

Participants are divided into two groups of equal size. Both groups follow the same course sequence with the second group starting a semester later than the first group. All courses are conducted in laboratory settings. Programming courses are taught in a laboratory of Apple IIe microcomputers. Since there are fewer computers than teachers, most teachers work in pairs during lab sessions. Elementary and middle school teachers from the District serve as project assistants and course teachers. Incentives include 15 free graduate credits (three graduate credits per course) and a $500.00 stipend for each participant who completes all five courses. In addition, participants are provided with free textbooks and supplies and as much computer time as needed and available in the Apple lab.

2.2 Preliminary Assessment.

Two groups of 26 participants were selected for this project from an applicant pool of 457. Forty-seven of the original 52 participants, or 90 percent, are expected to complete the project. On this criterion, the project has been more successful than believed possible.

There is also an early indication that the project will successfully motivate participants to seek positions as computing teachers at the elementary and middle school levels. Midway through the project, an official of the School District invited participants to apply for positions as computing teachers on a temporary basis. Approximately one third of the participants expressed interest and, within two months, several had been moved into such positions.

Other indications of the effectiveness of the current project are partial or anecdotal based on information available when the first group had completed all three programming courses and the second group had completed the two Logo courses. Some of the anecdotal information suggests that, although the project is achieving its goals generally, design modifications may be warranted.

Participants have performed well on programming projects, creating programs that are often quite innovative and enjoyable. Performance on midterm and final examinations also reveals that participants have a solid knowledge of programming concepts and of specific programming language features and a good understanding of how to apply this knowledge to solve simple problems.

In addition, participants appear to have mastered programming concepts and practices more quickly and completely than one would expect in a typical programming course. One of the BASIC programming instructors remarked that he had fewer problems teaching structured programming with BASIC to the first group than he usually has. This can perhaps be attributed both to the fact that they had been given a broad perspective and a top down view of computer programs during the first course and that programming was introduced using Logo which encourages a modular approach to program design.

First hand observations and informal discussions with participants and instructors during the programming courses confirm the anticipated benefits of the curriculum design. Preoccupation with programming language details does not appear to be as intense nor as extensive as had been the case in a previous teacher retraining project in which participants started to program in BASIC without any broad perspective about programming nor any real understanding of how programs function (Aiken & Snelbecker, 1990). On the other hand, some troubling behavior has been noted. It has been observed that participants are so highly motivated to succeed in the program and have such high expectations for their own performance that they are uncomfortable with less than perfect performance. This could be anticipated since participants were selected on the basis of high motivation to succeed. However, beginning and advanced Logo teachers have observed that these attitudes have been responsible for a significant degree of tension in the class. It has also been observed that participants inevitably become bogged down in programming language details, becoming frustrated and losing sight of the broader educational context in which these programming activities can provide stimulating learning experiences for children.

A somewhat more troublesome consequence of the participants' high self-expectations is the difficulty many have accepting and handling errors made while exercising new language constructs. As mentioned earlier, programming has been taught in a laboratory of Apple IIe computers with two teachers to a machine, encouraging continuous discussion and collaboration. As programming language features are introduced, participants are given simple exercises to perform which illustrate how these features work. In addition, participants are made familiar with general error recovery procedures and thus know that no error can be fatal.

During these lab sessions, discussion among participants often focuses in a nonconstructive way on errors, some participants becoming immobilized by simple mistakes. Many participants are unwilling to explore and play with new language features even when they are prompted with ideas for specific exploratory strategies. Perhaps participants have decided that the effort required to expore Logo features is not worth the trouble since such exploration seems irrelevant to completing assigned tasks or passing examinations. Another possibility is that they avoid exploring because they want to avoid making mistakes. Whether the structured focus of the course or the participants' high self-expectations best accounts for the observed error avoidance behavior, participants evidently have determined that exploration is not essential to learning to program with Logo.

In summary, objective indicators such as retention rates, and exam and project grades show that the model retraining program is an effective way to teach programming to experienced elementary and middle school teachers. However, anecdotal information suggests that the model may need to address more explicitly how teachers can learn to use programming knowledge to accomplish educational objectives in the classroom. This requires, minimally, a willingness to explore facets of programming languages without fear of making errors.

3. Teaching Teachers how to Teach Programming Effectively

One of the goals of the model retraining project for elementary and middle school teachers is to examine teaching strategies that maximize the general cognitive benefits of programming, such as development of problem solving and critical thinking skills that can be applied in both programming and nonprogramming domains. Objective measures of success used for this project do not directly measure the degree to which this goal has been achieved. Moreover, there is evidence that attitudinal factors may be interfering with its attainment. It is important to ensure that this educational goal can be met within the framework of the model program, and to develop ways of determining whether it has been achieved in practice.

3.1 The Problem.

As already noted, teachers selected to participate in the project are high achievers with exacting expectations for their own performance. While learning computer programming, a new and demanding subject for most of them, they have concentrated on mastery of the mechanics of the programming language, and many have become distracted by language detail, reluctant to explore new language features, and often appear tense. As is generally the case with introductory programming courses in universities, error avoidance behavior has been noted most commonly among participants with little or no previous computing experience, but tension and preoccupation with programming language mechanics are more prevalent.

These problematic attitudes and behaviors probably can be explained in terms of a combination of high self-expectation, lack of previous programming experience, and time constraints imposed by a full time teaching schedule. The problem is that these conditions cannot be changed under the constraints of the current retraining model.

Thus it is important to find other ways to help participants view errors as learning opportunities rather than as unpleasant obstacles and to minimize preoccupation with the mechanics of programming at the expense of general pedagogical objectives. Error avoidance behavior slows the process of learning how to program, but is even more worrisome in the case of future elementary school computing teachers for whom innovative, hence error-prone, approaches to programming will be an important educational tool. The primary objective is to prepare teachers to teach programming to children. Achieving this objective will require a good deal more than teaching the teachers how to program.

Ultimately, elimination of attitudes and behaviors that appear to interfere with learning may be insufficient. What is needed is some way to guarantee that the pedagogical goals of the project have been achieved. The problem is this: How can the difficulties of learning programming (often for the first time) be managed so as to permit the teaching experience of participants to be focused on the task of creating effective programming environments for children? Recent research on effective ways to teach and learn programming provide insight into how the project courses might be modified to meet this project objective.

3.2 Effective Programming Instruction Methods.

Through the 1970's, Seymour Papert and others claimed that children could experience general cognitive benefits from learning how to program (Papert, 1980). Papert advocated having children learn to program by freely exploring an interactive computer environment, a "microworld", whose simple objects and rules could be augmented indefinitely through programming. This "discovery" approach to learning to program contrasts with the standard approach where the instructor leads students through a structured syllabus. Papert believed that such an environment would enable programming expertise as effortlessly as living in France, say, enables fluency in French. Logo was designed to provide a discovery environment for learning programming.

By the mid 1980's, no conclusive evidence had been found to support claims that discovery learning with Logo promotes cognitive development any more effectively than do alternative instructional approaches, with or without programming (Pea & Kurland, 1984; Pea, Kurland & Hawkins, 1986). These results seemed to imply that little benefit would result from the sizeable investment in resources required to support teaching programming in elementary and middle schools. Recent research, however, indicates that more structured Logo environments, and, in particular, a mediated discovery environment, can facilitate childrens' mastery of programming and promote transfer of some problem solving skills learned in programming to other domains. In addition, several researchers have begun to describe characteristics of Logo programming environments that are likely to have specific cognitive benefits and to verify that these benefits do occur (Burns & Hagerman, 1989; Swan, 1989).

Recent studies of the cognitive benefits of programming with Logo compare unstructured (pure discovery), structured, and mediated learning environments. Each of these instructional approaches is laboratory based and includes initial presentation of new Logo content. Following content presentations, students in a pure discovery environment are free to explore Logo on their own, whereas students in a structured environment work on tasks assigned by the teacher. In the mediated discovery approach, instructors provide focused intervention and may assign or suggest some, but not all, tasks. Intervention is given to suggest problem solving strategies and to encourage development of metacognitive skills, helping students become aware of their own problem solving behavior, of its consequences, of alternative ways to achieve the same objectives, and of how problem solving strategies used in Logo programming may be applied to nonprogramming tasks.

Researchers are finding that general cognitive benefits can be achieved through programming only when children know programming well (Littlefield, Delclos, Lever, Clayton, Bransford & Franks, 1988; Pea & Kurland, 1984) and that structured methods seem to be superior to pure discovery methods for teaching programming well (Carver, 1988; Lehrer, Guckenberg & Sancilio, 1988; Littlefield et al., 1988; Swan, 1989). In addition, researchers are finding that mediated discovery methods, particularly those that include consideration of nonprogramming tasks, seem to achieve transfer of cognitive skills to nonprogramming domains more effectively than do either structured approaches, the pure discovery approach, or mediated approaches that are not computer-based (Lehrer et al., 1988; Littlefield et al., 1988; Swan, 1989).

A number of experimental studies have discovered measurable transfer of problem solving skills learned in programming with Logo to domains such as geometry, simple mapping problems, and error detection during a communication task (Au, Horton & Ryba, 1987; Carver, 1988; Lehrer et al., 1988; Littlefield et al. 1988). They also have demonstrated improved performance on measures of general problem solving skills, such as problem decomposition, error identification, and analogy (Keller, 1990; Swan, 1989). In addition, mediated Logo environments have been found to improve reflectivity (metacognition) and to foster divergent thinking (cognitive style) (Clements & Merriman, 1988).

Logo learning environments that are effective in developing general problem solving and metacognitive skills share the following characteristics:

(1) Sufficient time and explicit instruction is provided to ensure a mastery of Logo.
(2) Children are immersed in the problem solving process rather than the mechanics of programming.
(3) Instructors use focused intervention to encourage children to talk about problem solving processes in their own terms. These interventions use opportunistic questioning to prompt students to reflect on their problem solving processes (Au et al., 1987; Clements & Merriman, 1988).
(4) Explicit references are made to other domains such as ordinary planning activities or mathematics. Explicit references to familiar problems can reinforce mastery of problem solving skills in Logo and facilitate generalization to less familiar domains (Pea, 1989).
(5) Social interaction between teacher and child and between children is encouraged. Successful environments either encourage collaborative problem solving or schedule frequent discussions where children share accounts of the processes used to solve problems. Collaboration fosters conflict resolution and the discovery of better solutions. In addition, faster or more experienced children explicitly model reflective activity for other children. Such exchanges are collaborative rather than competitive (Clements & Merriman, 1988; Keller, 1990).

Transfer, which can be described in terms of generalizing from task specific situations to other knowledge domains, is fundamental to learning. Generalizable thinking skills have been successfully taught in mathematics (problem solving heuristics), in writing (planning and goal monitoring), and in reading comprehension (Pea, 1990). Learning environments that accomplish transfer in these domains share salient aspects of mediated discovery environments for Logo programming, namely, situated learning (knowledge and skills are taught in functional contexts), provision of multiple examples to help bridge to other domains, and development of metacognitive skills by modeling them and by encouraging students to explicitly emulate the model (Pea, 1990).

3.3 Reconsidering the Design of Programming Courses for Retraining Teachers.

Under the current teacher retraining model, programming is taught in laboratory courses that meet for three hours once a week. Language constructs and programming concepts are introduced in presentations which are interleafed with lab sessions where participants work on problems and projects assigned by the instructor. Thus programming is taught in a structured Logo environment as defined in the experimental literature. As noted in the literature, such environments seem to be more successful than unstructured environments at achieving programming mastery. The high standard of performance on examinations and programming assignments in project courses is consistent with these findings. On the other hand, mediated discovery environments appear to be as successful as structured environments at achieving programming mastery, yet more successful at achieving transfer.

These considerations suggest two alternative approaches to modifying the design of the project courses. Both approaches would place greater emphasis on instructional methods that foster development of general cognitive skills.

One approach would reduce the amount of time devoted to teaching programming languages by reducing the number of programming courses. Thus the program might be amended to teach one course each in Logo and BASIC or to eliminate the BASIC course. The remaining programming courses could then be followed by a course that explicitly treats cognitive issues in learning and explores the kinds of programming environments that tend to foster development of desired cognitive skills. Among other things, such a course could consider dialogue management, general problem solving strategies, and the embedded characteristics of programming environments that facilitate cognitive development (Burns & Hagerman, 1989) and enable transfer to related domains (Swan, 1989).

This approach would have the advantage of getting the details of the language out of the way so that participants could concentrate on pedagogical strategies in an instructional methods course. The disadvantage of this approach is that participants would have less programming instruction and so may fail to master programming sufficiently to feel comfortable teaching it.

A second approach to redesigning the programming courses would be to modify the existing courses so that they focus more explicitly on issues relating to how programming should be taught. This approach would not reduce the amount of programming instruction and so would not risk reducing the programming competency and confidence of participants. More significantly, it would allow teachers to experience the methods they would be learning, and later using to teach children. Since one tends to follow the behavioral models by which one learns, this approach is more likely to ensure that the new methods will be used.

Although problem solving strategies relevant to programming are emphasized in the courses as currently taught, participants still become preoccupied by programming detail at the expense of goals, strategies, plans, and overviews. A mediated discovery approach might help to counteract these inclinations. In such a mediated environment, teachers would be encouraged to monitor their own problem solving behavior and to monitor and guide the problem solving efforts of others (Clements & Merriman, 1988; Perkins, Schwartz & Simmons, 1988). They would also be encouraged to abstract problem solving strategies from the programming task (Littlefield et al., 1988).

The primary drawback of this approach is the effort required to teach it. It should be noted, however, that the equipment shortage which led to pairing participants on computers had unanticipated beneficial consequences. Participants often were able to help one another which allowed the instructor more time to engage in interaction with single participants or small groups. If pairing is done carefully to maximize collaborative interaction and to ensure that faster, or more experienced, participants are paired with slower, or less experienced, ones, then the instructor can assume more the role of a facilitator. One Logo instructor decided to change pairings half way through the semester. Although participant pairings had not been assigned they became set after the first few sessions and the instructor sensed that a few of these were dysfunctional because they reinforced non-constructive behaviors and attitudes. He had participants move anywhere in the lab so long as they had new partners. This change energized the class sessions noticeably.

4. Anticipated Benefits.

There are two reasons for modifying the program to include more explicit treatment of instructional methods and less treatment of the mechanics of programming. The first is that this better accomplishes one of the goals of retraining programs. The goals of any teacher training course in technology include both mastery of technology and of strategies for teaching the technology effectively. The revised curriculum would shift the focus from learning and learning how to teach programming languages to learning and learning how to teach general problem solving skills within a programming environment.

The second reason for focusing more on mediated learning strategies is to modify some of the attitudes noted among participants in the language courses as currently designed. The general idea is that by shifting the focus from mastery of programming to mastery of methods of instruction, the discomfort often experienced by novice programmers when faced with language detail can be managed.

This might be accomplished more effectively by integrating focused intervention techniques and peer collaboration into the programming courses and structuring these around problem solving strategies rather than programming language features. It seems likely that, having these courses focus on applying pedagogical skills which experienced teachers already have will alleviate the learning difficulties that have been observed, particularly the observed reluctance in many cases to explore the Logo environment. It also seems likely that instructional strategies that have been found effective in developing programming mastery for children can have similar benefits for teachers in a retraining program.

It should be emphasized that project courses so far have satisfied the objective measures of success for the model program and, in some cases, have far surpassed expectations. It is suggested that changes in the structure of the programming courses may contribute further to the model program's general effectiveness.

Acknowledgements

The work reported here has been funded under National Science Foundation Project Grant TPE-8850531.

References

Aiken, R., & Snelbecker, G.E. (1988). *A Model Program for Retraining Elementaryand Middle School Teachers to Use Computers Effectively in the Classroom.* Proposal for project funded by the National Science Foundation. Philadelphia: authors.

Aiken, R., & Snelbecker, G.E. (1990). Hindsight: reflections on retrainingsecondary school teachers to teach computer science. *Forthcoming in Journal of Research on Computing in Education.*

Au, K.W., Horton, J., & Ryba, K. (November 1987). Logo, teacher intervention,and the development of thinking skills. *The Computing Teacher,* 12-16.

Burke, W. (1990). Information technology in the 5 to 16 curriculum -a crosscurricular theme. In McDougall, A., & Dowling, C. (Eds.), *Computers in Education: Proceedings of the IFIP TC 3 Fifth World Conference on Computers in Education-WCCE 90,* Sydney, Australia, July 9-13, 1990 (pp. 31-36). Amsterdam: North-Holland.

Burns, B., & Hagerman, A. (1989). Computer experience, self-concept and problemsolving: The effects of LOGO on children's ideas of themselves as learners. *Journal of Educational Computing Research, 5* (2), 199-212.

Carver, S.M. (1988). Learning and Transfer of Debugging Skills: Applying TaskAnalysis to Curriculum Design and Assessment. In Mayer, R.E., (Ed.), *Training and Learning Computer Programming: Multiple Research Perspectives* (pp. 259-297). New Jersey: Lawrence Erlbaum Associates.

Clements, D.H., & Merriman, S. (1988). Componential developments in LOGOprogramming environments. In Mayer, R.E., (Ed.), *Training and Learning Computer Programming: Multiple Research Perspectives* (pp. 13-54). New Jersey: Lawrence Erlbaum Associates.

Goodspeed, J. (1989). The NEA Publishes two powerful reports on educationaltechnology. *Electronic Learning, 9* (1), 21.

Haney, M.R. (1990). Infusing technology into the high school curriculum -a fiveyear report from the Blair Magnet Program. In McDougall, A., & Dowling, C. (Eds.), *Computers in Education: Proceedings of the IFIP TC 3 Fifth World Conference on Computers in Education - WCCE 90,* Sydney, Australia, July 9-13, 1990 (pp. 109-113). Amsterdam: North-Holland.

Keller, J.K. (1990). Characteristics of Logo instruction promoting transfer oflearning: A research review. *Journal of Research on Computing in Education, 23* (1), 55-71.

Lehrer, R., Guckenberg, T., & Sancilio, L. (1988). Influences of LOGO onchildren's intellectual development. In Mayer, R. E., (Ed.), *Training and Learning Computer Programming: Multiple Research Perspectives* (75-110). New Jersey: Lawrence Erlbaum Associates.

Littlefield, J., Delclos, V.R., Lever, S., Clayton, K.N., Bransford, J.D., &Franks, J.J. (1988). Learning LOGO: method of teaching, transfer of general skills, and attitudes toward school and computers. In Mayer, R.E., (Ed.), *Training and Learning Computer Programming: Multiple Research Perspectives* (pp. 111-135). New Jersey: Lawrence Erlbaum Associates.

Papert, S. (1980). *Mindstorms: Children, Computers, and Powerful Ideas.* NewYork: Basic Books.

Pea, R.D., & Kurland, D.M. (1984). On the cognitive effects of learning computerprogramming. *New Ideas in Psychology, 2* (2), 137-168.

Pea, R.D., Kurland, D.M., & Hawkins, J. (1986). LOGO and the development ofthinking skills. In Pea, R.D., & Sheingold, K., (Eds), *Mirrors of Minds* (pp. 178-197). Norwood, N.J.: Ablex Publishing Corp.

Pea, R.D. (April 1989). Socializing the knowledge transfer problem. Institutefor Research on Learning, Report No. IRL89-0009.

Perkins, D.N., Schwartz, S, & Simmons, R. (1988). Instructional strategies forthe problems of novice programmers. In Mayer, R.E., (Ed.), *Training and Learning Computer Programming: Multiple Research Perspectives* (pp. 153-178). New Jersey: Lawrence Erlbaum Associates.

Swan, K. (1989). Logo programming and the teaching and learning of problemsolving. *Journal of Artificial Intelligence in Education, 1* (1), 73-92.

Special Presentation
Technology! Calculus! and Individual Differences!

Brent Wilson, Michel Pitre
Media Integration Program, Carleton Roman Catholic School Board
Nepean, Ontario

Technology! Calculus! and individual Differences! is an exciting classroom. It involves the integration of computers, tablets, videos, text, and software into every calculus course in all our high schools. It has received worldwide recognition by educators and school boards.

Developed by the Carleton Roman Catholic School Board, this exciting new approach to teaching and learning calculus includes sets of three software packages per unit, for use in conjunction with traditional and other teaching approaches. All software components generate questions randomly in such a way as to allow literally billions of variables, while using virtually no disk space! Tutorial sections are interspersed with detailed illustrations, animations, examples, highly visual interactive investigations and simulations.

These classrooms are designed to enhance individualized learning through the integration of high technology and multimedia products into the curriculum. It places curriculum and high technology products into the hands of the regular classroom teacher, thereby enabling the teacher to meet curriculum objectives, different learning styles, and rates of learning within the classroom environment . Student record keeping and tracking are facilitated through the software components.

As part of an overall school board initiative, these classrooms demonstrate the power and benefits of technology in the future of education.

Special Presentation—ISTE

Accreditation Standards for Instructional Computing and Technology Teacher Preparation Programs

This session presents the standards and guidelines developed by the International Society for Technology in Education (ISTE) Accreditation committee for evaluation of programs in instructional computing and technology. These guidelines address the standards and criteria for unit accreditation and folio approval that will be submitted for review by the National Council for Accreditation of Teacher Education (NCATE) in September 1991.

Research providing the basis for development of the standards includes competencies for preparation of all teachers, administrators, and candidates in programs at the Bachelor, Master, and doctoral levels. results of the study and the process teacher education programs follow when seeking accreditation for their programs in instructional computing and technology are discussed and examples of activities meeting the standards given. The panel of presenters including members from the NCATE/ISTE Liaison, and ISTE Accreditation Committee are available to answer questions regarding the standards and guidelines for unit and program accreditations.

Panel Members:

Don Knezek
Accreditation Committee
Education Service Center—Region 20
San Antonio, TX

C. Dianne Martin
ISTE/NCATE Liason
George Washington University
Washington, DC

Sally Sloan
Accreditation Committee
Winona State College
Winona, Minnesota

Harriet Taylor
Accreditation Committee
Louisiana State University
Baton Rouge, Louisiana

Lajeane G. Thomas
ISTE Representative to NCATE Board
Louisiana Tech University
Ruston, Louisiana

James Wiebe
NCATE Board of Examiners
California State University, L.A.

Special Presentation

The Texas Center for Educational Technology: Current and Future Solutions

Kenneth E. Brumbaugh
James L. Poirot
Texas Center for Educational Technology

The Texas Center for Educational Technology was created in June, 1990, by the Texas Education Agency at the University of North Texas, in collaboration with the University of Texas at Austin. The major goals of the Center are to:

- create a statewide exchange program for scholars, teachers, researchers, and industry personnel;
- expand summer in-service teacher training programs for using technology in the classroom;
- create model classrooms to demonstrate the latest in technology-based teaching and learning;
- have the nation's strongest educational computing and technology-based research effort.

The Center's eleven research and development laboratories cover the following areas:

1) Assessment of Student Learning and Cognition
2) Student Learning and Special Populations
3) Teacher Productivity and Training
4) Teacher Retention and Support Networks
5) Curriculum and Instruction
6) Instructional Design and Evaluation
7) Educational Telecommunications and Informatics
8) Multimedia and Emerging Delivery Systems
9) Design of Computer-Based Instruction
10) Brain Mapping and Evoked Potential
11) Collaborative Software Development

TCET serves the Texas Education Agency, foundations, community colleges, coalitions, universities, professional organizations, businesses, and school districts statewide. It will research new technology applications for education, create prototypes of educational applications for technology developed for commercial or other purposes, and develop classroom applications for current technology.

Projects
Technology and Special Populations

The CLC Technology Project: Meeting Literacy Needs of Special Populations

Jane D. Steelman
Carolina Literacy Center
Louisburg, NC

Abstract

The Carolina Literacy Center, created under a grant received by the Department of Medical Allied Health Professions at the University of North Carolina at Chapel Hill, is the first facility of its kind at the state or national level to focus attention on the literacy needs of individuals with severe speech and physical impairments. (SSPI).

The center is committed to addressing the literacy needs of the more than 10,000 SSPI individuals in North Carolina. An estimated 90 percent of such individuals read at levels significantly below their nondisabled peers. The achievement of literacy opens doors to academic and vocational advancement for these individuals.

The project presentation will describe the technology project and its stage of development at the time of the presentation.

Center for Special Education Technology

Susan Elting
The Council for Exceptional Children
Reston, VA

Abstract

The Center for Special Education Technology at the Council for Exceptional Children is a national information center funded by the U.S. Department of Education, Office of Special Education Programs. Through its active participation and development of vital information products, the Center has become a national leader in the area of special education technology.

CENTER SERVICES

As a national resource with the goal of increased use of information about technology in the education of children with disabilities, the Center provides theme-related and general information services.

Theme-Related Services. A major component of the Center's services is its in-depth focus on identified themes in special education technology. theme activities address expressed information gaps by identifying existing knowledge and organizing that knowledge in various product formats. To date, five information themes have received Center attention.

The Center's efforts in assistive technology focus on products and services that provide access within the educational/learning environment. In the area of funding, information products emphasize awareness of funding sources and strategies for the acquisition of technology in educational settings. The technology training theme emphasizes the infusion of technology in the process, content, and products of preservice and inservice training. Technology integration focuses on factors that contribute to the successful integration of technology in educational systems. Technology for the moderately handicapped student explores the potential of technology-assisted instruction for this population.

General Information Services. To meet the continuing need for general awareness information about the use of technology and alternate resources in the field, the Center provides ongoing general information services.

Current products include:

State and topical Resource Inventories listing local, state, and national technology resources that provide information and services for professionals and consumers. A Tech Use Guide series summarizing issues and applications in technology use such as computer Access, Technology for Preschool Children, and Computers and Cooperative Learning. the Marketplace is a periodic publication highlighting vital marketing issues affecting special education technology. TECHLINE is the Center's electronic newsletter on special Net which provides current and topical information to special education professionals and education agencies.

SUMMARY

The Center's information products are targeted for service providers and product providers active in the use of technology in special education. These audiences in turn provide information and support to the technology consumers—educators, parents, and individuals with disabilities—who ultimately apply the technology in living and learning. The Center's products are in the public domain and are routinely made available to representatives of target groups and to individuals on request.

I Care; You Care: Together, *You* Can Do It!

Diana M. Willis
Suwannee District Schools, Live Oak, FL

Abstract

This project centered around youth in a group child care facility for dependent, neglected, and abused children. These youth are, on the average, two academic years behind their peers. This is one of the many reasons these youth are missing the incentive to learn and to excel. These "at-risk" youth are in danger of failing to complete their education with an adequate level of academic and interpersonal communication skills.

The project used computers, software written especially for "at-risk" youth, extensive counseling sessions, and a pre/post attitude survey.in an effort to foster through positive learning experiences the growth of both self-esteem and adequate levels of academic skills. It was hypothesized that by building self-esteem, attitudes toward school and behavior toward peers would change measurably.

The project was partially funded through a grant from the Suwannee Foundation For Excellence In Education, Inc. and through in-kind contributions of other business partners. The presentation will share information concerning the final status of this phase of the project.

Paper

The Nihongo Tutorial System for Learning Technical Japanese

Nelson K. Leung
Anthony A. Maciejewski
Purdue University

Abstract

The Nihongo Tutorial System (NTS) is an Intelligent Computer Aided Instruction (ICAI) system designed to assist western scientists and engineers in developing a reading competence of technical Japanese. The goal of the NTS is to maximize comprehension through context by providing individualized lessons that conform to the student's technical interests. It makes use of an augmented parse tree data structure to store syntactic, semantic, phonetic, and morphological information about an article, and incorporates a student model to represent the student's interests and current Japanese proficiency. In addition, the tutorial system supplies several electronic dictionaries for ease of cross-referencing. Through the NTS, the student is able to learn the general concepts of usage, meaning, and pronunciation of Japanese text as well as the style of technical Japanese writing.

This material is based upon work supported by the National Science Foundation under Grant Number INT-8818039. The Government has certain rights in this material.

Introduction

Although interest in learning Japanese is increasingly growing, many English-speaking people find it very difficult to learn the language due to its complexities. The situation is even worse for those western scientists or engineers who would like to read technical publications published in Japanese because of a number of factors. First, there are very few textbooks available to illustrate the style of technical Japanese writing. Second, textbooks usually do not cover specific technical disciplines in detail. In addition, the student cannot rely on using Japanese technical journals for self-study because only a small percentage of such publications are also available in English. Third, there are very few instructors who are fluent in both Japanese and English in addition to having expertise in a technical area. Finally, since most learners of technical Japanese are already engaged as professionals, they cannot spare a great deal of time and effort attending a structured class to study the language. To address these problems, a computer based tutorial system is an effective solution because the computerized tutor can automate most of the instructional tasks and provide relevant instruction to meet the diverse needs of technical Japanese students.

The Nihongo Tutorial System (NTS) described here is an Intelligent Computer Aided Instruction (ICAI) system developed to assist people in developing reading competence of technical Japanese. The goal of this tutorial system is to maximize comprehension through context, and to minimize rote memorization and dictionary use. The NTS is composed of three components: the Nihongo Tutor, the Parse Tree Editor and the Administrator. The Nihongo Tutor helps the student to learn the general concepts of usage, meaning, and pronunciation of Japanese text by providing technical Japanese articles for study. It also keeps a record of the student's current Japanese proficiency level in his personal database. The Parse Tree Editor, which is the authoring tool for this ICAI system, is used to prepare information about each article for the tutor by incorporating syntactic, semantic, phonetic, and morphological information into the parse tree of each sentence. The Administrator performs automatic article selection for the student by assigning an appropriate article that matches his current Japanese language proficiency and his technical areas of interest. This presentation focuses on a description of the Nihongo Tutor. A detailed description of the Administrator and the Parse Tree Editor can be found in the literature.[1]

The NTS was developed to work on the Macintosh line of computers due to their desirable human-machine interface and cost effectiveness. This also allows it to take advantage of the available Japanese Interface System, called *KanjiTalk*, to assist in the input, display, and editing of Japanese characters.

Overview of the Japanese Writing System

The Japanese writing system is composed of two distinct orthographies known as Kanji and Kana. The Kana are further divided into Katakana and Hiragana. Kanji are logographic characters, originally adopted from the Chinese language, for writing the vast majority of Japanese words such as nouns, verbs, and adjectives.[2] An example of a Kanji compound is 日本語 (nihongo), meaning the "Japanese language." Although there are more than 50,000 Kanji characters in existence, only about 2,000 of them are designated by the Japanese Government as official "Jōyō Kanji" for daily use.[3] These Jōyō Kanji account for approximately 90% of the Kanji found in standard texts.

Each Kanji has two kinds of readings: the "on" and "kun" readings. A single Kanji may have two or more readings, each indicating a different meaning for the character. For instance, the kun reading of 語 is かた(る) (kata(ru)), which means "talk, relate," whereas its on reading is ご (go), meaning "word" in Japanese compounds.[4] Each Kanji consists of a series of strokes written in specific directions. The stroke count, 14 for the character 語 as an example, is one of the traditional methods to categorize Kanji. Most Kanji characters can be divided into smaller recurring components. Over 90% of Kanji have two parts: the radical indicating the general area of meaning of the character, and the other representing the phonetic component. The radical of the character 語 is the component on the left-hand side, 言. A common way to search for a Kanji in a dictionary is to use its readings, stroke count, radical, or a combination of the above.[5]

Unlike the Kanji, the Kana are phonetic symbols developed to represent Japanese syllables. The meanings of the original characters are ignored. The Japanese devised the two separate syllabaries, Hiragana and Katakana, to simplify the process of phonetically writing Japanese text. They are two different scripts for writing the same sounds. For example, writing 日本語 phonetically in Hiragana results in にほんご. Katakana are mainly used to transliterate words borrowed from foreign languages whereas Hiragana are used to write native Japanese words such as inflectional endings of conceptual words like verbs, particles, auxiliary verbs, and also words that are primarily grammatical in function.

Learning to read Japanese is not an easy task if one considers all of the different aspects of the language such as the large number of characters, the large number of homonyms, the lack of word delimiters, the complexity of character images, and the complex lexical principles used to combine the three character sets. Many of these problems complicate the usage of Japanese in computers as well. However, the Japanese have devised a standard set of computerized Japanese characters known as the Japanese Industrial Standard (JIS) to allow for the display and use of Japanese in computers. The Japanese have also developed methods to transliterate Roman characters into Kana characters as well as for converting Kana characters into Kanji.[6]

Intelligent Computer Aided Instruction

Given that the Japanese language is so difficult for Westerners to study, the NTS has been developed to make use of computer resources to assist instructors in teaching technical Japanese. It is an intelligent CAI system that uses knowledge representation and deduction about the domain and each student to control most aspects of the pedagogical interactions with little assistance from an instructor.

A CAI system usually contains two parts: authoring system software and delivery system software. The authoring system software provides a system by which courseware can be created or edited. The delivery system, on the other hand, controls the interaction between the student and the tutorial system by providing subject material, recording the student's responses, and so on.

Three important components of an ICAI system are the representation of the subject material, student modelling, and the instructional strategy used. ICAI is usually targeted towards complex learning domains so that the instruction programs are flexible and generative (that is, they can be run repetitively by the same student although his learning competence is different each time). In NTS, instruction is individualized by using a student model to account for the student's knowledge and capabilities.[7] By storing information about each individual student (such as his current proficiency level, his areas of expertise and interest, etc.), NTS selects relevant subject material and relates his responses to his competence model. The instructional strategy used by NTS manages to organize tasks like topic selection or student evaluation into an intelligent process so that the most appropriate lesson will be assigned to a student.

Thus far, very few computerized instructional aids have been developed for learning Japanese. Those that have been described in the literature include the New Direct-Reading Japanese-English Character Dictionary[8] by Jack Halpern, KanjiCard[9] by Kazuko Nakajima and KanjiMaster[10] by Dezso Benedek. These systems are designed primarily to teach people to understand more about Kanji, providing extensive information about the meaning, reading, usage, and graphic features of individual Kanji characters. In contrast, NTS puts more emphasis on Japanese syntax, the contextual use of Kanji, the correlation between the Kanji and Kana, and showing the style of technical Japanese writing. Thus the philosophy of NTS is comparable to that of the Japanese CAI project at the University of Pittsburgh.[11]

The Nihongo Tutor

The Nihongo Tutor is the delivery system software of the NTS. Its underlying task is to enhance the student's proficiency of technical Japanese by providing him Japanese articles to study. The student is expected to learn the general concepts of usage, meaning, and pronunciation of Japanese text as well as the style of technical writing through the tutor. The purpose of the tutor is to optimize comprehension through context, and to minimize dictionary lookup and mechanical translation. The article assigned is carefully selected according to the student's current Japanese competence, the level of difficulty of the article, and his technical area of interest. Presumably, if the student does not know the meaning or pronunciation of some portion of the text's contents, he will request semantic, syntactic, or phonetic information from the tutor. Consequently, the tutor uses a parse tree data structure to represent the subject material in order to provide fast and efficient interaction between the student and the tutor.

The tutor also furnishes several electronic on-line dictionaries for cross-referencing purposes: a character dictionary, a Kanji compound dictionary, a grammar dictionary and some specialty dictionaries. The tutor keeps a record of what the student learns from the current lesson in order to maintain and support student modelling. It stores all material that the student masters in addition to that with which he has difficulty. New information is stored into the student database whenever he finishes studying a lesson. This kind of database management helps the system to assign an appropriate article for him as the next lesson.

Representation of the Subject Material

For natural language processing, each sentence in an article can be represented by a parse tree based on the syntax of the language. The semantics of the sentence is also augmented into the parse tree as additional information stored in each node. Figure 1 shows how a Japanese sentence is represented in a parse tree. Note that each node in the parse tree may have an arbitrary number of child nodes.

A generalized tree is a suitable data structure to represent the augmented parse tree. In the NTS, this generalized tree is implemented as a doubly-linked list. The linked list can effectively connect all nodes together for fast searching and retrieval. Moving upward or downward along different levels of the parse tree is accomplished readily due to the double link between the nodes. This allows moving downward along a parse tree in order to show how a sentence or a phrase is broken down into its constituents, in addition to moving upward to depict how several clauses are combined together to form a sentence. The double link thus allows the student to investigate Japanese sentence structures more easily. This is especially important when learning Japanese since there is no character used to denote word or phrase boundaries. Thus, it is sometimes difficult to understand the Japanese sentence structure because the first step of breaking a sentence down into phrases is not trivial.

Original Japanese sentence:

速度というのわ速さと向きで表わされるものである。

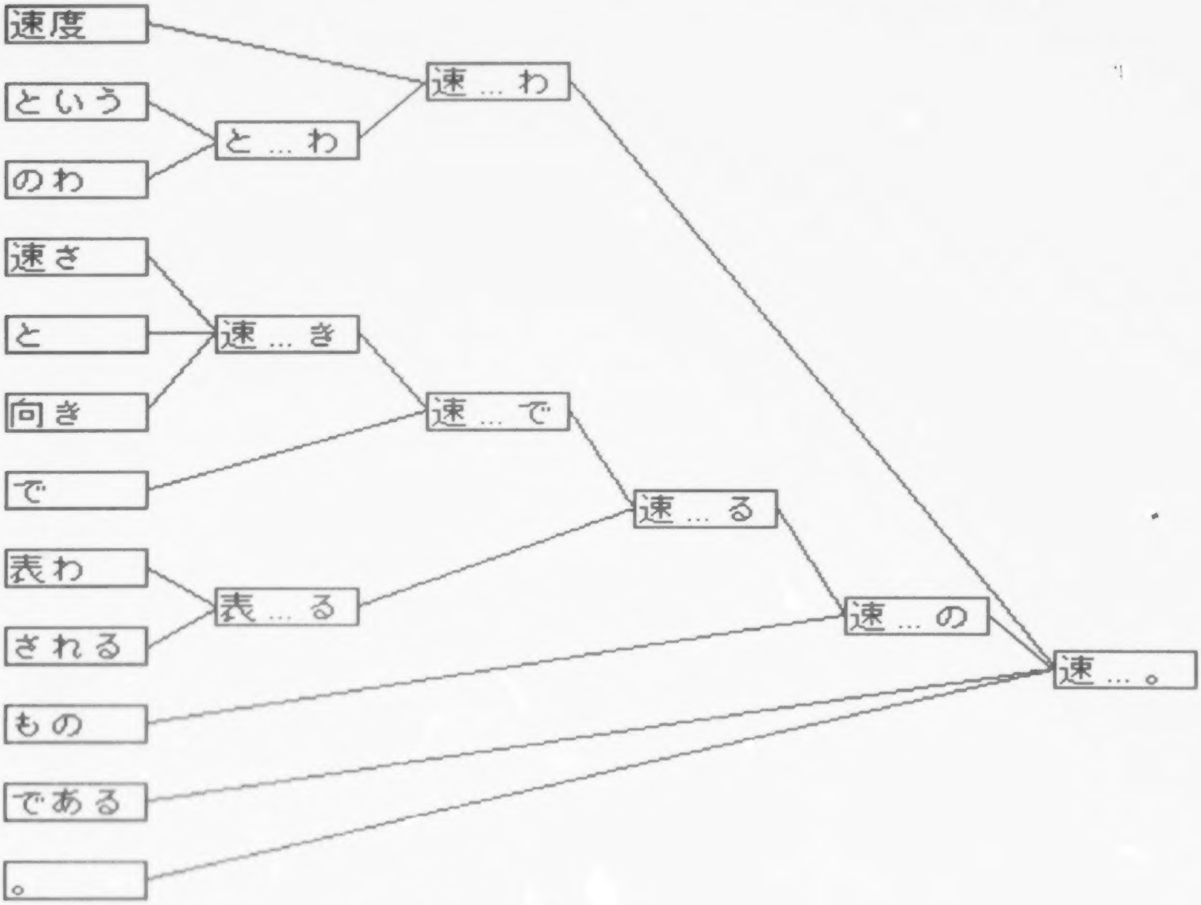


Figure 1. A Parse Tree Representation of a Japanese Sentence

With the help of this parse tree data structure, translation information is retrieved by the scheme shown in figure 2. Whenever the student has difficulty in understanding some of the text in the article, he highlights this text to notify the tutor. To get the translation information, the Nihongo Tutor locates the highlighted text in a sentence and returns the associated parse tree. It then finds a node in the lowest possible level of the parse tree enclosing all of the selected text. All the translation information based on this node is retrieved and then displayed.

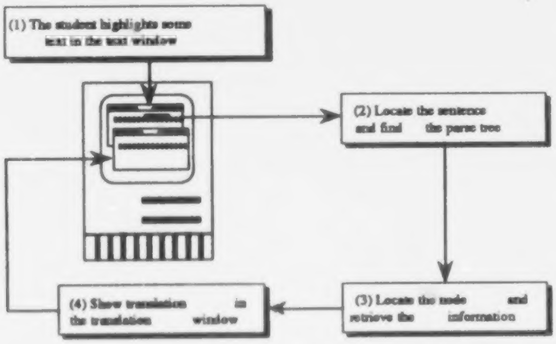

(1) The student highlights some text in the text window
(2) Locate the sentence and find the parse tree
(3) Locate the node and retrieve the information
(4) Show translation in the translation window

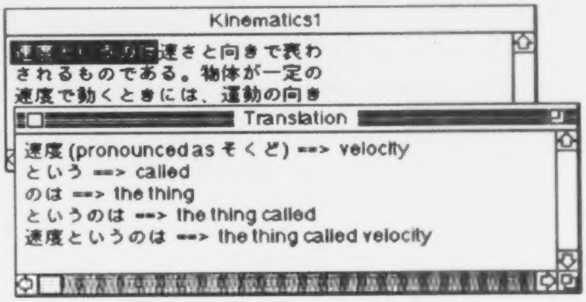
Figure 2. Interaction between the User and the Nihongo Tutor

The attributes of each node in the parse tree incorporate all semantic, phonetic, and morphological information, as well as any other supplemental data for instruction, in a compact form in order to remove any redundancy. In particular, they include the lexical boundary of the stored text, its best English-equivalent contextual meaning, Japanese readings or pronunciations, possible syntactic breakdown, and so on.

Syntactic Analysis

The tutor provides a utility, called syntactic analysis, in order to assist the student in learning about the grammatical rules of Japanese. This is accomplished by showing how a clause or a sentence is split into its basic constituents. With the help of this kind of breakdown, the student can understand more about the syntactic structure of a Japanese phrase, in addition to its overall meaning.

Figure 3 presents an example of syntactic analysis. The task is achieved by moving upward along the parse tree. Going to the parent of a node automatically includes all the sibling nodes. To show how several words or phrases are combined together to form a clause, the tutor uses a double-click facility together with the Macintosh mouse. If a double-click occurs when the mouse pointer is on some highlighted text, the selection range will be changed to that of the parent node. This happens because a node in the upper level of the parse tree is being selected. Similar double-clicks will move the selection upwards until the root node is reached. If syntactic analysis is requested, the translation information of all the descendant nodes will be displayed. This is designed to enhance the student's comprehension of word or clause boundaries in a Japanese sentence.



速度というのは速さと向きで表わ
されるものである。物体が一定の
速度で動くときには、運動の向き

速度 (pronounced as そくど) ==> velocity
という ==> called
のは ==> the thing
というのは ==> the thing called
速度というのは ==> the thing called velocity

Figure 3. Example of a Syntactic Analysis

Electronic Dictionaries

In Japanese, it is often the case that there will be more than one meaning associated with a Kanji. The exact meaning in an article depends on the context of its usage. The general contextual meanings of Kanji can be acquired by comprehending usage instead of by rote dictionary memorization. Ideally, the design of the tutor is such that use of the dictionaries will be minimized. However, if a Japanese word is used in several articles with various meanings, then an electronic on-line dictionary is valuable as it provides the student a correlation between the different usages of the word.

The set of electronic dictionaries in the NTS includes a character dictionary, a Kanji compound dictionary, and even several specialty dictionaries, built up with specific technical terms from various scientific disciplines. In the current version of the implementation, the character dictionary contains 1,945 entries, including the basic Jōyō Kanji, along with their most important readings and definitions. The compound dictionary currently incorporates 1,000 common Kanji compounds which are the most frequently occurring compounds appearing in Japanese newspapers and books.[12] The specialty dictionary is composed of a group of special dictionaries, each devoted to a specific technical discipline. For instance, the specialty dictionary in the tutor now contains a communications dictionary encompassing 8,889 entries of communication terms. In addition to the character, compound, and specialty dictionaries, there is also a grammar dictionary with 346 grammatical entries.[13]

The goals of the electronic dictionary can be summarized as follows:

(1) To provide the student with an understanding of the meanings, readings and characteristics (stroke counts, radical

number, Nelson's dictionary reference number) of Kanji and Kanji compounds.

(2) To enable the student to look up Kanji characters and compounds with high speed and different search schemes.

(3) To facilitate a cross-reference between the translation in an article and the definitions in the dictionary.

Each entry in the character dictionary includes a Nelson's dictionary reference number, a radical number, a stroke count, common meanings of the character and their associated readings. The Nelson's dictionary reference number is an index to *The Modern Reader's Japanese-English Character Dictionary* by Andrew Nelson, one of the earliest Japanese-to-English dictionaries which many students use for reference. Since a Kanji has many properties, the student is allowed to use different searching criteria to cross-reference Kanji characters with similar characteristics. For example, when none of the readings of a Kanji are known and determining its radical number seems too troublesome, the character can be located by means of its stroke count.

For each dictionary search, all meanings and readings related to the search key are displayed. The possible access keys are by Kanji, readings (on and kun), Nelson's dictionary reference number, radical number, stroke counts, and English translation. The English translation key offers a method for searching Kanji characters based on their English definitions. The Nelson's dictionary reference number, the stroke counts, and the radical number of a Kanji are all unique; however, this is not the case for its on and kun readings.

The Kanji compound dictionary consists of common Kanji compounds, including their meanings and the associated pronunciation. The search key is either the entire Kanji compound or one of its individual Kanji. If the latter is provided for searching, all relevant compounds are retrieved so that a cross-reference can be obtained. The grammar dictionary has an English translation, a description of the grammatical structure and one or more examples for each entry. Each example contains a Japanese sentence illustrating the usage of the entry, followed by a sentence showing the English translation for that Japanese sentence.

Student Model

To monitor the student's progress in Japanese proficiency, the tutor records important performance information to aid the Administrator in deciding the student's competence. Such information includes material that the student knows well, as well as those items not currently mastered. This information is in terms of Japanese words, phrases, and even general grammatical structures. Each time the student highlights some Japanese text and requests translation information, the tutor updates his personal database so that he will be presented with articles appropriate for his level of competence in the next lesson.

One of the difficulties in building the student model of language proficiency is evaluating why a student has asked for a particular piece of information. One direct way to evaluate the student's competence is to query the student to see if he has difficulty with a phrase whenever he requests translation information. However, this is too time-consuming and would interrupt the process of instruction in order to gather accurate data for assessing proficiency. Thus, the trade-off of not collecting such data to maintain the flow of study is tolerated. The evaluation scheme used in the tutor is to classify textual material in an article into two areas: well-known and unfamiliar. The first one keeps a record of all material that the student has mastered whereas the second one records those items that he has been exposed to yet still has difficulty comprehending. In order to classify material into either the well-known or unfamiliar area, the tutor makes the assumptions that the student requests translation information for some Japanese text if and only if he does not understand it. As a result, this material will fall into the unfamiliar area and any other material in the text will be classified as well-known. Clearly, these assumptions can be violated in practice; however, it is a reasonable approximation to the way that most students behave.

Since a Kanji may have various readings in different contexts, it is important that these readings are recorded to reflect what readings the student currently knows. Whenever he learns new readings of a word, they are added to update his database. Each entry in the unfamiliar area has a frequency counter which is a measure of the difficulty that the student is having with understanding this entry.

Student Evaluation

In practice, there is no clear-cut division between material that is known or unknown so that these databases should not be independent of each other. For instance, if a student learns some new Japanese text, it is initially put under his unfamiliar area. After a few lessons which include the same material, the tutor should transfer it from the unfamiliar area to the well-known area if the student does not once again ask for the translation. On the other hand, any material in the well-known area should be moved to the unfamiliar area if the student requests information on this material. Hence the evaluation scheme accurately reflects the student's change in competence by handling both the case where the student learns more about Japanese as well as the case where he forgets some text that he has studied before.

The evaluation scheme in the NTS adds, updates, or transfers entries to the unfamiliar area whenever the student asks for a translation. Any text for which the student does not request information is added or transferred to the well-known area after he finishes reading the lesson. There is no guarantee that this material is necessarily well-known by the student; however, putting it under this area does not give rise to a problem. If this material really should fall into the unknown area, the scheme will transfer it once the student requests translation information in future lessons. Thus, this scheme resolves conflicts so that the same Japanese text in different lessons would not be put into different areas. The records which hold the information for the known and unfamiliar area are loaded into the system in the form of a hash table data structure in order to support rapid access to this large amount of data.

The selection of an appropriate article for the student is performed by the Administrator software which makes use of the student model along with information about the student's technical interests. In particular, it compares the student's personal database with the lexemes of the article, (i.e., all leaf nodes in the parse trees of the sentences), in order to give a rough estimation of how much new material will be presented to him. An example of information used for article selection is displayed in figure 4. The instructional strategy is to assign an article that has a specific amount of new material, thus precluding unnecessary repetition of already mastered materials and maximizing comprehension through context.

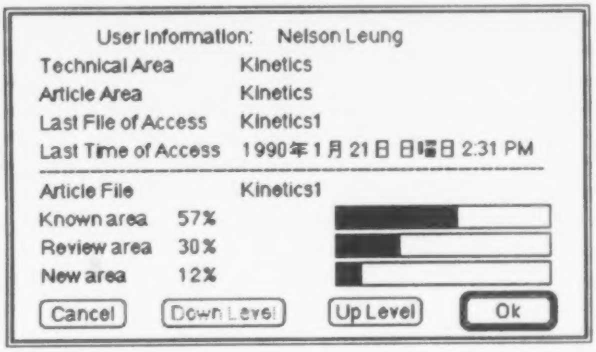

User Information: Nelson Leung
Technical Area Kinetics
Article Area Kinetics
Last File of Access Kinetics1
Last Time of Access 1990年1月21日 日曜日 2:31 PM
Article File Kinetics1
Known area 57%
Review area 30%
New area 12%

Figure 4. A Comparison of a Student's Database with the Lexemes of an Article

Conclusions and Future Work

As previously stated, the main goal of the NTS is to assist scientists and engineers in developing reading competence in technical Japanese. The system is currently being tested at a selected number of universities. Future work includes further enhancement of the NTS such as the incorporation of more electronic dictionaries and more dictionary entries, automatic translation assistance of technical Japanese articles, and the preparation of a rich set of instructional texts.

Bibliography footnoted in text

1. Leung, K. W. (1990, May). *The Nihongo tutorial system for learning technical Japanese*. Masters Thesis, School of Electrical Engineering, Purdue University, West Lafayette, IN.
2. Yasuo, Y. (1973). *Japanese for today*. Japan: Gakken Company, Ltd.
3. Unger, J. M. (1987). *The fifth generation fallacy*. New York: Oxford University Press.
4. Hadamitzky, W. & Spahn, M. (1981). *Kanji and Kana*. Vermont: Charles E. Tuttle Company, Inc.
5. Nelson, A. (1962). *The modern reader's Japanese-English character dictionary*. Vermont: Charles E. Tuttle Company, Inc.
6. Payne, P. & Utsuru, Y. (1989). *MacKanji 6.0 user's manual*. Washington: Linguist's Software, Inc.
7. Rickel, J. (1989, January/February). Intelligent computer aided instruction: a survey organized around system components. *IEEE Trans. on Systems, Man, and Cybernetics*, 19(1), 40-57.
8. Halpern, J. (1985, December). New direct-reading Japanese-English character dictionary. *CALICO Journal*, 3-6.
9. Nakajima, K. (1988, December). On developing HyperCard stacks for the study of Chinese characters: KanjiCard. *CALICO Journal*, 75-87.
10. Benedek, D. (1990, April). Computers for reading and writing Japanese. *42nd annual meeting of the Association for Asian Studies*, Chicago, IL.
11. Nara, H. (1990, April). Issues in developing a Japanese CAI. *42nd annual meeting of the Association for Asian Studies*, Chicago, IL.
12. Kikuoka, T. (1970). *Japanese newspaper compounds*. Vermont: Charles E. Tuttle Company, Inc.
13. Makino, S. & Tsutsui, M. (1986). *A dictionary of basic Japanese grammar*. Japan: The Japan Times, Ltd.

Production note:

As you'll note, some of the text in this paper appears different from the usual text styles used throughout the *Proceedings*. This is because parts of this paper came to us in a special Japanese operating system called *KanjiTalk 6.0*, used with *MacWrite*. The result is that the sections containing Japanese characters are reproduced as bitmapped graphic; hence, the different appearance. We thought this distinction interesting, and left it as you see it here for your information.

Ian Byington
ISTE production assistant

Paper
"What did he say?": Speech Output in Preschool Software

Erik Strommen
Children's Television Workshop
New York

Abstract

Advances in technology have made it possible to include speech in electronic learning materials for small children. Little is known, however, about how young children respond to speech as a component of technology. Based on the results of a series of studies of Sesame Street interactive learning games tested with over 150 children aged 3 to 5, guidelines for evaluating speech as a feature of computer software are presented. Preschoolers responded best to speech segments that were short, contained concrete language, and were not stereotyped or repetitive. Information presented solely via speech was much less effective than when speech segments were combined with supplemental, task-related visual information that reinforced their content. Guidelines derived from these results can be used for judging the quality and effectiveness of speech included in interactive technologies such as computer software and videodiscs.

Recent and rapid changes in technology have made speech a possible feature of interactive electronic learning materials. To date, "talking machines" have most commonly been studied as aids in reading and writing tasks (c.f. Parham, 1988; Reitsma, 1988; Weiner, 1991). However, the potential uses of speech in applications for small children, not as an adjunct to task content but as an effective form of interface, have been given very little study. Because young children cannot benefit from onscreen text messages, spoken requests and feedback can potentially empower young children's learning in complex tasks that currently require verbal guidance and explanation by an adult. Such uses of speech as an interface, however, must be carefully considered. There is abundant evidence that young children have difficulty decoding spoken language in everyday contexts (Beal and Belarad, 1990; Hargrove and Panagos, 1982; Markham, 1979), so issues of how best to integrate speech into interactive multimedia technologies must be carefully considered.

The selection of the nature and content of speech in computer systems is especially important because unlike human speakers, who can tailor their speech to their listeners in an endless variety of ways, the affordable computer of the near future will typically only be able to repeat a limited variety of stock phrases. How long these phrases should be, how they should be worded, and where they should be placed as part of the interactive task, are issues of vital importance if speech is to be used as an effective, engaging component of educational materials for pre-literate children.

The present paper summarizes the results of a series of seven formative studies on computer learning materials for young children that employ speech as an integral component. The goal of this summary is to describe how children reacted to speech as a component of the computer activities, and to make a first attempt at identifying some guidelines for the optimal length, content, and placement of speech in interactive learning materials. These guidelines can serve designers of software using speech, but more importantly they can be used by teachers and others who need to evaluate new products for their suitability with their children.

Method

Participants

One-hundred and fifty children ranging in age from 34 to 62 months, drawn from the greater New York area, served as subjects. Equal numbers of boys and girls were tested.

Materials

Prototypes for seven educational software games for preschoolers, using Sesame Street characters and settings, were used in the present study. These games were designed to teach a variety of familiar preschool skills, such as rhyming, simple arithmetic, etc. Each was identified with a specific Sesame Street character who served as the game's "host" in an animated setting similar to familiar Sesame Street settings (Ernie and Bert's house, Big Bird's nest, etc.). The scripts for the games were written based on conventions from the Sesame Street television series, and were designed to cover the possible alternatives that crop up in interactive tasks: initial instructions, feedback messages, prompts, and end-of-game comments. The spoken portions of the games were recorded by the actual Sesame Street characters, and then incorporated into the games via digitized speech systems available on standard IBM DOS computers.

Six of the games were designed with the same broad format. To play the game, children placed a cursor on the muppet and pressed a button to start game play. The character then explained the game (the muppet animated during spoken segments), and play commenced. In four of the games, the spoken instructions contained crucial game information (words to find rhymes for, attributes to match with objects, verbal math problems, etc.); in two others, the spoken feedback from responses contained the crucial information. All six of these games also relied almost exclusively on spoken dialogue from the characters to convey incorrectness/correctness of responses, repeat key information, and prompt the child. The response structure was also the same for all six games: The child was required to move the cursor to objects and select them as the basic method of game play. If, for example, Ernie asked the child to "Find things that are blue," the child would move the cursor to an object and click on it; a correct response received a reward animation and Ernie saying "My airplane! It's blue! Good going!" and a randomly varied prompt encouraging the child to find another object that is blue. The child would go on to click on the next object, and so on.

The seventh game tested was slightly different, and involved the child "phoning" different muppets and listening to verbal segments reinforcing basic counting, addition, and initial-sounds concepts.

Procedure

The children, with an adult Researcher as "helper," all played multiple rounds of each game, with each child playing only one game each. The children's interactions with the games were videotaped, and responses to the games, especially to the verbal materials, were scored for the presence of confusion or hesitation, need to repeat spoken segments, requests for assistance from the adult, and affective reaction. The children's spontaneous comments were also recorded.

Results and Discussion

While the results vary somewhat depending on the particular game being described, six consistent findings relating to the use of speech recurred across all seven of the games tested.

1. It appears that instructions lasting 20 seconds or less, and feedback messages and prompts lasting less than ten seconds in length, are ideal. Children tended to ignore, or fail to attend to, important content in a long speech stream. While adults had no trouble picking up the important information, the children often missed it entirely, responded only to the last few words, or reacted only to very simple, concrete language such as "...press the button!". Such performances may reflect a bias by young children toward expecting that communicative messages are meant to prompt an action (Ackerman, 1981; Shatz, 1978).

2. Children tended to respond to the computer as if the character was actually speaking to them, particularly when action words or commands were spoken. If the character queried "Are you ready?" children invariably answered "Yes!" as if they could be heard. Such prompts proved to be very beneficial for cuing children that important verbal information was about to be presented, as did the prompt "Watch this!" for cuing the child to important visual information, and "move your cursor to..." for prompting game action.

3. The children responded to the inflection of the language as much as to the content of what was said. Exuberant language on correct response feedback, such as "That's right!" or "Hooray!" was highly reinforcing for children, and low-key or questioning affect ("My hat?") was effective for feedback to an incorrect response. It was also found that negative feedback, given exuberantly, was often misinterpreted as implying correctness even though the actual content of the speech belied the inflection.

4. Spoken prompts stating the names of objects as they are highlighted can facilitate game play. In several of our games, children were required to select objects based on their name (finding a number six, locating a rhyming object, etc.). Having the object's name spoken if the child stops the cursor on it for a few seconds proved to be an excellent way to support the performance of children who could not identify the objects in question.

5. Children are sensitive to stereotyped or repetitive speech. Children frequently asked "Why does he always say that?" or "I know already!" after a few rounds of game play. They often begin to talk over, or ignore, phrases they had heard repeatedly even if they did not understand the content of the message. Having speech-based systems employ a variety of phrases for the same situation (having four varied error messages, for example) would be beneficial.

6. Visual cues supplementing verbal material, particularly in tasks where the child is required to remember key information, enhanced game appeal and useability. We found that simply presenting problems verbally was not nearly as effective as combining this material with redundant onscreen information. Children drew on both modalities in recalling the information they needed, and used the visual cues as a reminder when they accidently forgot or did not attend to the verbal material.

Conclusions

The use of speech in interactive products constitutes a new area for educational psychologists and teachers. The findings from the present studies suggest several concrete criteria that can be used to determine the effectiveness of speech as a part of preschool software. They are applicable to all interactive media, not just computer software; any multimedia technology (videodiscs, interactive compact discs, etc.) intended for preschool use that employs spoken language can be evaluated using the following guidelines.

1. The length of the spoken segments in the program should be brief. Long, expository segments tax young children's listening abilities and attention spans.

2. The language used in the program should be as concrete as possible, and should direct the child to undertake a specific action. For example, if the task is to match animals with their appropriate sound, an ambiguous error phrase such as "Try it again," should be avoided; a phrase like "Pick a new animal," is better.

3. The speech should be as natural as possible, using appropriate inflection wherever possible. Positive affect in messages such as "That's right!" and neutral or questioning affect on wrong answers seem to enhance children's ability to understand the program, as well as help motivate task performance.

4. Programs employing a varied number of messages for the same situation are preferable to those with only one, repeated message. In our studies, children quickly tired of repeated phrases and came to ignore them, defeating their purpose.

5. The language in the program should be supplemented by visual material. The speech segments should have a concrete relationship to objects or events happening on the screen. Such redundancy between spoken and visual elements helps children to "figure out" what to do, if they do not fully grasp the verbal material. In our math game, for example, the verbal problems are actually performed as they are verbally described. This allows children who may initially fail to process the spoken material to nonetheless play along, and it also encourages children to connect the language used to the situation at hand.

References

Ackerman, B. (1981). Performative bias in children's interpretations of ambiguous referential communications. *Child Development, 52*, 1224-1230.

Beal, C. R. & Belarad, S. C. (1990). The development of message evaluation skills in young children. *Child Development, 61*, 705-712.

Hargrove, P. M. & Panagos, J. M. (1982). Role saliency cues and children's sentence comprehension. *Journal of Psycholinguistic Research, 11*, 217-228.

Markman, E.M. (1979). Realizing that you don't understand: A preliminary investigation. Child Development, 48, 986-992.

Parham, C. (1988). Computers that talk. Classroom Computer Learning, 8(6), 26-31.

Reitsma, P. (1988). Effects of guided reading-while-listening and independent reading with computer-based speech feedback. *Reading Research Quarterly, 23*(2), 219-235.

Shatz, M. (1978). On the development of communicative understandings: An early strategy for interpreting and responding to messages. *Cognitive Psychology, 10*, 271-301.

Wiener, R. (1991). Computerized speech: A study of its effect on learning. *T.H.E. Journal, 18*(7), 100-102.

Special Presentation

Software —From Research to Development to Publication

Barbara Kurshan, Chair
Educorp Consultants, Roanoke, VA

Software development for education is occurring at a variety of levels and at a dramatic pace. As the market demands more computer and multimedia instructional products, the consumer and the developer has to be able to determine if the product is "valuable" and "viable." A recent education study group has defined software of value when it is used consistently and it can be employed as a vital element to improve teaching. While software is viable if it survives in use long enough to enable courses to be changed by the software. This panel will address the process of researching and developing valuable software that publishers will think is viable. The presenters will explore the future of software in the K-12 and collaborative higher education market from the perspective of funder, developer, and publisher. In addition, the session will address issues such as how to approach funding sources, working with developers in your classroom and taking your product to a publisher. Topics covered will include:

- Obtaining funding for software development research
- Current research based software projects
- New software and multimedia development
- How to get development projects published
- The next generation of software

Panelists:

Marge Cappo
Wings for Learning, Inc.
Scotts Valley, CA

Steve Ehrmann
Program Office for Interactive Technologies
Annenberg/CPB Project,Washington, D.C.

Peter Kelman
Scholastic Inc.
New York, NY

John Richards
Software Development Manger,
BBN Laboratories Incorporated
Cambridge, MA

Special Presentation—ISTE

Hypermedia in the Classroom

Ray Braswell
Auburn University, Alabama

Mark Horney
University of Oregon

This session will discuss and display classroom materials and projects developed by teachers and researchers in the area of hypertext, hypermedia, and multimedia.

Projects

History/Social Studies and Hypermedia/Multimedia or Interactive Media

Using Hypermedia to Implement Curricular Change

Linda Smith, District Technology Specialist
Sheri Henderson, Mentor
Saddleback Valley Unified School District, California

Abstract

After 18 months, Saddleback Valley Unified School District's Hypermedia Project has resulted in teachers authoring 65 original Hypermedia stacks in at least 13 areas of curricular reform. Hypermedia is the linking of media via the computer. This school district of 25,000 students in Mission Viejo, CA specifically chose to link the laser videodisc player to the Apple Macintosh computer using *HyperCard* software. Thirty-nine teachers from 29 schools were given the training and resources necessary to include Hypermedia in the curricular and instructional reform process.

Sheri Henderson, one of our presenters, will demonstrate her simple yet powerful *HyperCard* utility program which allows anyone to create interactive, multimedia lessons using any videodisc; users merely copy, paste, and type. Examples from her project—which integrates American literature, art, and history—will show its ease of use and flexibility.

This Hypermedia presentation was produced through the Project to share our successful strategies for districtwide implementation of multimedia in all subject areas. Attendees will receive handouts on selecting participants, training, budget, follow-up, increasing publicity opportunities, purchasing, and recommendations for multimedia software and hardware.

Civil War Interactive Multimedia Project

Charles White
George Mason University, Virginia

Abstract

The proposed project presentation will report on the status of the Civil War Interactive Multimedia Project, a design and development project currently funded by the Corporation for Public Broadcasting and being carried out by the Center for Interactive Educational Technology at George Mason University in Fairfax, Virginia.

The project is based on the PBS series The Civil War, a documentary by Ken Burns of Florentine Films, Aired on five consecutive nights in September of 1990 Burns' series achieved the highest ratings in the history of public broadcasting. The task facing the interactive multimedia designers has been to capture the power of the video material while broadening and deepening the learning opportunities through the power of an interactive multimedia environment. The presentation will focus on four key design challenges and how we have met those challenges.

The first challenge has been to design a product that will address some of the serious deficiencies in social studies knowledge and critical thinking skills reported in recent national assesments and in the reform literature. Knowledge of the Civil War is marginal among high-school seniors; indeed, some 68 percent of them could not place the Civil War in the Correct 50 year period. knowledge of fundamental principles of government underlying the war are also not well understood by today's students. Based on extensive examination of the literature, we derived general principles of history, key historical themes, and knowledge, skill, and value goals for the project that we believe target some of the most glaring gaps in student understanding.

Another challenge has been to glean the most useful and effective video, audio, and textual data from Ken Burns' considerable archives. Burns had a 14-to-1 shooting ratio for his 11-hour broadcast series, yielding some 150 hours of video material to be reduced to two sides of a videodisc. Thousands of still photographs and hundreds of pages of diaries and documents have been at our disposal, and we will describe the procedures we adopted to manage the selection task.

A third challenge has been to provide maximum flexibility for exploration while providing sufficient structure to guide student inquiry in a rich knowledge base. the designers believe that students with a wide range of learning and cognitive styles should be successful users of interactive multimedia. We will discuss some of the difficult trade-offs required to achieve what be believe is an appropriate balance of free-wheeling investigation and guided inquiry.

Finally, the design team has been focusing on teacher training throughout the project. We will discuss a multi-dimensional strategy for teacher training built into the design of the videodisc and print materials as well as ancillary video-and print based training materials.

Given the limited amount of presentation time, detailed information relating to our discussion will be delivered via handouts.

Teaching History/Social Science Through Interactive Technologies

Nancy Hunt
California State University, Fresno, CA

Abstract

We live in a media-rich society, surrounded by television, radios, movies, and digital sound. Satellite communication makes possible the local publication of a nationwide newspaper and allows us to see and hear events as they happen around the world.

If we are to make school an exciting and relevant place to be, we must reach our students through, and empower our students to use, the technological tools of the age. Unfortunately, the typical classroom is less well equipped than the average American den.

This presentation will describe a project which addresses this technology-gap between the typical home and school environment. In the fall of 1990, faculty from CSU, Fresno and the Clovis Unified School District were awarded an IBM equipment grant to develop history/social science lessons and staff development materials which use interactive laserdisc technology.

This presentation discusses the process by which the teachers were introduced to laserdisc technology and assisted as they created their interactive lessons. Lesson excerpts will be presented and any changes evidenced in teacher and student attitudes towards the content area and the use of technology in the classroom will be discussed.

Special Presentation

Exploring Simulations in Science

John Richards
Bolt Beranek and Newman

Yoel Givol
Logal Educational Software

Bill Barowy
Maine Maritime Academy

Mark Rodriquez
Lexington High School, Lexington, Massachusetts

Simulations bridge the gap between complex mathematical theories and experience, and are changing the very nature of scientific inquiry. They provide new visual phenomena that aid in building scientific intuitions. Scientists are developing new, investigative skills, and asking novel, searching, questions.

Science teaching can now capture these exciting developments and reflect the way scientists presently work. Understanding processes is easier when students can explore the dynamics and nature of the changes made. Students can further their knowledge when they are given the opportunity to become actively involved in altering a process, and predicting and observing the effects of their modifications. by questions and the skills to seek informative answers.

This panel will discuss their experiences with the Explorer system: an interactive environment for exploring simulations that gives students and teachers access to new tools that are designed to provide students with these experiences. The Explorer provides.

- Simulations: taking advantage of the computer's ability to present interesting, interactive animation. Dynamic, graphic simulations in physics (mechanics, light, waves, electricity, and modern physics) and biology (genetics, heart, population, photosynthesis).

- Analytical tools: providing access to multiple related representations of the data. The spreadsheet, graphs, histograms, and numerical displays update in real time while the simulation runs. Graphs are drawn or data sampled to the spreadsheet as you view the simulation

- Scripting: provides facilities to design labs, experiments, and lessons. For example, you might create a button for graphing the energy of a system as a function of time, or write a script defining a condition to detect a collision.

Special Presentation—ACM/SIGCUE

Cooperative Learning - Computers and Technology

Greg Brownell, Nancy Brownell
Bowling Green State University

This session will be an overview of cooperative learning strategies and methods for integrating cooperative learning into computer education. A report will be given of a research study on cooperative learning.

Practical implications for teachers will be discussed, and materials will be made available.

Projects
Teachers and Telecommunications

CAWP Online: Supporting Professional Growth with Electronic Linkages

Jane A. Rosen
Instructional Coordinator Pritzker School
Chicago Board of Education

Abstract

The presentation will discuss a project which electronically links Chicago Area Writing Project's summer institute with each other and the Writing Project Director. CPS installed an EIES network in September, 1989 but it was initially underutilized by local educators and was not directly tied to district curriculum and instruction objectives. The investigators connected participants from the well established Chicago Area Writing Project with the exciting Bulletin board System as a economical effective, and efficient means of facilitating greater support for and interaction between Writing Project teachers, a secondary goal was to publicize the project and gain support for expansion. Moreover, they intended to apply the model they developed for teachers of writing to other curricular purposes.

In this session, the presenters will describe the following aspects of the project in the handouts and remarks during the session:

- the process of obtaining support, approval, and assistance from CPS central office and building administrators;
- procedures for initial on-line training of Writing Project teachers
- the on-going technical and curriculum support system provided to Writing Project teachers
- preliminary findings on the effectiveness of this project
- efforts to publicize and gain support for the project within the Chicago Public Schools
- future plans to refine and expand this program and to extend the concepts to support and sustain staff development in other curricular areas

Teacher Inservice Technology Education:
A Partnership Approach in California.

Craig Blurton
Director, California Technology Project
Seal Beach, CA

Abstract

The California Technology Project is a cooperative effort of the California State University system and the California Department of Education to provide support services for technology-using K-12 and preservice teachers educators. This session will focus on the free statewide telecommunications network, the Technology Resources In Education (TRIE) electronic telecommunications network, the Technology Resources In Education (TRIE) electronic information service, maintained by the project for use by educators.

Special Presentation

Virtual Realities: From the Concrete to the Barely Imaginable

Stephen Marcus, Graduate School of Education
University of California, Santa Barbara

This session deals with "deep media". If text *tells*, and video and film *show*, then virtual realities *embody* the worlds they create. One seems to enter such simulations physically. These environments involve unusual interfaces and metaphors for future educational computing. Included are systems like *Reality Built for Two, the Virtual Environment Workstation, Mandala*, the *Alternate Reality Kit*, and *A Very Nervous System*. The presentation includes a discussion of virtual reality hardware and software and explores the educational implications of radical shifts in point of view (e.g., one can *become* the data), of being responsible forwhat someone else sees, and of functioning in environments in which space itself is the interface.

Projects
HyperTraining

HyperMedia: A Performance-Based Model of Inservice Education

Marty Karlin
Trinity Center, California

Abstract

The State of California Curriculum Frameworks are calling for teachers to incorporate active, thematic, and cooperative learning methodologies across the curriculum. Students are to learn to obtain and value knowledge, reach conclusions and develop sound judgement. This inservice project seeks to make the training, tools, materials, and support available to help teachers and students meet those requirements.

As 30 teachers in 17 rural California schools learn to access and employ interactive multimedia technologies in their classrooms, they become exposed to Information Age teaching and learning environments that are intuitive and versatile, and have great appeal. Critical factors have been administrative support, curricular integration, equipment availability, ease of use, training, and individualized follow-up. This report will include a description of the rationale, goals, expectations, training activities, and results after two years of acquisitions and inservice. Teaching methodologies, effects on learning, and teacher/student products will also be discussed.

HyperTEACH- A *HyperCard* Teacher Utility

Michael Land
Midwestern State University, Texas

Abstract

In an effort to get preservice and inservice teachers to begin to use *HyperCard* to develop instructional stacks with a minimal amount of training, the presenter prepared a teacher utility stack called HyperTEACH. Students can begin to prepare their own stacks after only a few hours of practice with HyperTEACH. With HyperTEACH, an individual can prepare an instructional stack that randomly displays some type of encouragement after correct responses, and indicates when the user has selected incorrect responses, and returns them to the original question. The presenter will share some of the stacks his students have prepared using HyperTEACH. Individuals at the session will be given copies of HyperTEACH that they can use with their students.

HyperCard for Elementary and Secondary Education: A Sampler

Kevin Walsh
Connie Gonser
John H. Strange

Abstract

For the past two summers the school district of Royal Oak (Michigan) has conducted a one week, 40 hour course in HyperCard for teachers in the district. almost half of the teachers who attended the HyperCard training had never used a computer before. All the teachers produced products, many of which have been or could be, developed into instructional programs for use within the district. This session will enable participants to see a sample of the teacher produced products, to receive a diskette of that sampler, and to discuss the strategies that we have followed in encouraging additional teacher training throughout the year.

Special Presentation

Two International Telecommunication Models: I*EARN (The International Education and Resource Network) and the New York State/Moscow School Telecommunications Project

I*EARN was launched by the Copen Family Fund, Inc. in 1989. It is a pilot model project that is exploring the impact of telecommunications internationally on teaching methodologies, student learning and international understanding. The purpose of the telecommunications is to have students work together, internationally, on projects which they design - projects which will, in some way, make a meaningful contribution to people and the planet. I*EARN is an expansion and international application of the New York State/Moscow School Telecommunications Project.

At this writing, there are 5 Centers in the US that can act as regional models for the demonstration and replication of this work. The Centers are:

1. The GEM Project in Yorktown Heights, New York.
2. The ELNA Project in Boston, Massachusetts.
3. The Century 21 Project in Seattle, Washington.
4. The Irvine School District - as part of the Education for the Future Project in Irvine, California.
5. The Albuquerque Public Schools.

The five US Centers of I*EARN are connected by telecommunications to the USSR (Leningrad, Moscow, and Kiev), Israel (Jewish and Arab Schools), the Netherlands and Argentina. In the next few months it is expected that schools in China (PRC), Spain, the Phillipines, Indonesia, Finland and England will join the network. Discussions are in process with schools in Mexico, Costa Rica, Egypt, France, and Japan.

The New York State/Moscow School Telecommunications Project is the first of its kind between the US and the USSR. In June 1988 an agreement was signed between the Copen Family Foundation (working in collaboration with the NYS Education Dept.) and the Soviet Academy of Sciences to initiate a 3 year pilot project to assess the impact of telecommunications and exchanges on teaching methodologies, student learning and international understanding.

This project links 13 schools in Moscow with 13 schools in New York State by telecommunications - electronic mail and video speaker telephones. Student, teacher and administrator exchanges support the relationships between the schools.

The purpose of the school linkups by telecommunications is to have students engage in projects of their own design - projects which can in some way make a contribution and be of service to people and the planet while meeting academic requirements. The purpose of the exchanges is to further support the student generated projects as well as increase international understanding.

At this writing, several hundred students, teachers and administrators have participated in telecommunication projects and exchanges (home stays, primarily) and produced some remarkable results.

Since the purpose of this work is to create a model for replication by others, the Project presentation will focus on a brief summary of the work, highlights of the successes, an articulation of the problems and an open discussion of issues about replication.

Panelists:

Peter Copen
The Copen Family Fund, Inc.
Yorktown Heights, New York

Ed Gragert
The Copen Family Fund, Inc.
Yorktown Heights, New York

Alexander Uvarov
The Soviet Academy of Sciences

Jane Barley
NYSED

Paper

Ending the "Blind Leading Blind" Syndrome: Advocating Technologically Competent Administrators

John F. Beaver
SUNY College at Buffalo, New York

The late 1980s witnessed the rise of many crucial social issues that captured the national spotlight and cried loudly for immediate solutions. An increased emphasis on drug education, prevention of teen pregnancy, AIDS awareness, and other current concerns are causing school administrators to shift their attention, and their funds, from instructional computing to other programs. As Mary Jo Langhorne, et al, suggest: "The use of computers in education is not the compelling issue it was five—or even two—years ago. New bandwagons have presented themselves: Higher order thinking skills, merit pay plans, and objective-based education now compete for the attention and funding computers received but a short time ago."[1] Although educational computing has received strong public support in the recent past, many new topics are monopolizing media attention.

During the past three years, many schools have had fewer resources available to them. When "belt tightening" takes place, some programs receive insufficient support. How have instructional computing programs fared in this environment of financial instability? In 1987, I conducted a national study of outstanding elementary school instructional computing programs. One disturbing finding was that the schools' computer budgets were very volatile, with few schools having established computer expenses as permanent line items in their school budgets. Kathleen Fulton reports that the Office of Technical Assistance found similar tendencies: State-level profiles indicate that state support is subject to the volatility of economic conditions, and of political and educational priorities in the individual states. Another pattern to emerge from the Beaver study (1987) was that funding levels for instructional computing programs had declined dramatically during the late 1980s. The findings indicate that many administrators felt they'd already paid their dues by purchasing some necessary hardware. They were now prepared to allow other important educational issues to take financial precedence. For the most part, they lacked long-range plans to guide the steady development of their programs.[2]

In addition to the fiscal instabilities, some sources continue to challenge the importance of using computers educationally. A recent *New York Times* headline states: "Computers are in the classroom, but no one is paying much attention to them." A headline in the *Wall Street Journal* reads: "Computers Failing as Teaching Aids." The popular media suggest a reduction of interest in educational computing at a time when more support is needed to ensure maintenance of programs. Langhorne, et al, point out, "...there are those who are already concerned about the waning interest in instructional computing and the decline in financial support for microcomputers." This confused situation promises programmatic chaos unless school leaders provide guidance and support. The leaders must develop clear understanding of instructional computing program needs so that they can develop long-range plans that insure those needs are met. Unfortunately, too few school leaders have the necessary computer competence, and consequently, far too few comprehensive plans exist.

A recent National School Board Association survey of American school districts concluded that there is very little evidence of comprehensive planning for educational computing programs. The survey findings conclude that most district plans "did not have a comprehensive flavor." Why is this true? Developing a comprehensive, long-range plan that will support an educational computing program requires a leader with a deep understanding of the problems and promises of instructional computing. *Many* of our administrators simply do not have this understanding. Greg Kearsley contends that "administrators often lack any suitable preparation to effectively manage the computing activities in their schools."[6] I agree; our administrators are being asked to make decisions on a topic for which they are not prepared.

If we expect our administrators to provide the vision and understanding needed to guide the development of instructional computing programs, we must encourage them to increase their computer competence. As Sylvia Charp argues: "It is generally agreed that more still needs to be done in faculty development if we expect truly broad use of technology in teaching and learning to occur. This should include programs designed for administrators of all levels, to enhance their awareness of technological literacy and their competence in planning for technology utilization."[4] The administrative programs Charp references should not be limited to inservice experiences, but should include college level instruction that constitutes a portion of the required program for state administrative certification.

Kathleen Fulton points out that only "...18 states require...that all students in their teaching degree programs or those seeking certification take a course on [educational] computer topics."[5] About one-third of our states value educational computing enough to mandate it in their teacher education programs. And, if less than one-third of the states require computer education for teacher certification, far fewer require a similar course designed for administrators. Still, without such experience to help administrators develop the computer competencies they need, there will be no clear vision guiding instructional computing programs.

Many educational leaders have called for greatly increased computer training for our teachers. For instance, as Robert McCarthy quotes Tom Snyder: "We all made a serious mistake in giving the computers to the kids first. We ignored the teachers, cut them out of the loop, and that's precisely the way to kill a promising educational technology."[9] Len Scrogan includes other limiting factors, but agrees with Snyder's main point: "Why isn't technology having a greater impact? Many factors are to blame: insufficient access to computers, inadequate supplies of quality software, and pauper technology budgets all contribute....[but] the most significant factor lies in the fact that the vast majority of teachers today have little or no training in the use of the new technologies."[11]

In general, even teachers are still not receiving the preparation they need to use technology successfully. Dave Moursund points out: "By and large our colleges of education are doing a miserable job of preparing teachers to deal with the Information Age."[10] Granted, teachers *do* need more assistance with using technology instructionally. But who is in the best position to insure that the necessary logistical, financial, and psychological support that teachers need is provided for them? Their local administrators, of course, if the administrators understand the unique need for support that instructional computing adoption requires.

Certainly, empowering our teachers—raising their level of technological competence—is very important. I strongly endorse greatly increased staff development efforts to raise teachers' technological understanding and comfort level. However, if the promises of educational technology are to be kept, the education of our administrators is equally—perhaps more—important. Somehow, when we started the computer revolution we forgot to invite the administrators .

Too often, unfortunately, administrators have no hands-on experience or quite limited hands-on experience with microcomputer experience. Without that personal experience, they can't really understand the need for continuing support that instructional computing demands. Administrators have to develop the experiential base they'll need to guide their instructional computing programs. One part of that foundation includes the hands-on experiences that a course on administrative uses of microcomputers can provide.

Computer educator David D. Thornburg argues strongly for the need for informed leadership: "Educational computing took its sabbatical last year, and now it's time to get to work, recharged and reinvigorated with energy and enthusiasm. Yes, additional hardware costs more money, but that's not the problem. The problem, as I see it, comes from a lack of vision."[12] Although I agree with Thornburg, I suggest that the road to vision includes journeys through the "slough of frustration"—lost files, confusing screen displays, unclear support documentation, printer malfunctions, hardware and software incompatibilities, information overload—to the "highlands of empowerment"—polished document production; facile data manipulation, swift numerical analysis, powerful instructional delivery. If administrators have no personal experience with microcomputers, they can't really understand the training and support that teachers need to help them learn to master the machines. Without that understanding, developing educational computing vision is not possible.

These points were brought home sharply to me this summer when I taught an educational administration course titled "Administrative Uses of Microcomputers" at SUNY College at Buffalo. My experiences parallel those reported by Kearsley when he taught a similar course three thousand miles away. "Many of the administrators who take the course have had no prior experience with computers and in some cases are antagonistic or anxious about them."

The course I taught is an elective in the educational administration certification program. Course participants are aspiring administrators, individuals seeking administrative certification as a precursor to finding an administrative position. Nearly half of those enrolled in the two sections had *no* previous computer experience. None! Some were not even familiar with a keyboard.

Most of those without experience shared another characteristic: they were afraid of computers. My experience supports Andrew Barbour's findings from a national survey of administrators: "17% of the administrators cite fear of computers among their staffs as a major problem facing their wider implementation in their school or district."[1] Although Barbour doesn't mention it, I suspect many administrators share the fear of technology with their staffs. That fear is a fear of the unknown. And that fear, as with all fears of the strange or unusual, dissipates with understanding and familiarity. Those qualities only come with experience.

The course proved especially rigorous for those starting from scratch, designed as it was to provide a broad range of experiences and insights to participants. However, after completing the course, the participant consensus was that the hands-on expectations provided a critical part of their experience. If they hadn't had the opportunity to master the computer for themselves, they felt that they would not have understood the powers and the problems of technology adoption. When they eventually receive administrative appointments, these leaders-of-the-future will not be guilty of "dumping" hardware on untrained teachers. They understand the need for continued, substantial support, both financial and psychological.

The course includes practical applications, group discussions, individual and co-operative learning experiences, and opportunities to explore in depth a particular application of choice. The applications include the productivity enhancement standards: word processing, data base managing, print merging, spreadsheet developing, chart generating, information interchanging between applications. The power of each application is discussed, with special attention given to how that power applies to the role of an educational administrator.

Each application is also taught during class sessions, using Microsoft's *PC Works* because of its power, ease of use, and flexibility. Next, students practice using the application independently while completing assignments which represent typical administrative uses of that application. For instance, once they learn word processing, students are asked to produce an "opening-of-the-school-year" letter that demonstrates appropriate use of the powers provided by word processing. The powers mastered include changing font style, font size, font type, spacing, and justification.

After data base management is taught, participants practice it by creating a student information file. Then they demonstrate using the data base program's searching, sorting, and reporting powers by printing out reports appropriate for different situations. The next computer use introduced is print merging—combining word processing and data base powers. Students demonstrate their print merging proficiencies by generating personalized form letters that satisfy on-going administrative communication needs such as parent open house communication, parent conference letters, or field trip announcements.

Once spreadsheets have been mastered, class members use them to create budgets or inventories for their schools. Next, they use the figures from their spreadsheet budgets to generate graphic forms using *Works*' charting powers. Finally, they combine their understanding of word processing, data bases, spreadsheets, and charting by sharing data from one application component to another. These integration powers let students develop polished word processed reports that include spreadsheet figures, data base records, and charts.

In addition to the practical applications that focus the course, class participants also engage in varied group discussions and problem solving experiences. These situations provide more pervasive opportunities for applying their computer understandings. Key topics included in course discussions are deciding administrative computer uses, establishing optimal computer adoption approaches, creating stable computer budgets, determining computer placement, planning instructional computing programs, planning staff development of computer competencies, planning computer integration policies, and developing cost-effective policies for computer usage.

Beyond the course's individual and class requirements, each participant is asked to work in a cooperative partnership. This task demands a joint approach to learning a new technology and presenting the pertinent issues it introduces to the rest of the group. Each individual forms a dyad with another class member. Each pair is expected to become familiar with a "new technology" topic of agreed-upon choice and report on its administrative implications to the class-at-large. Suggested subjects that the teams are encouraged to investigate and report on include applications of telecommunications, uses of dedicated input devices (such as scanners and digitizers), the costs and features provided by dedicated applications (like scheduling and attendance programs), the administrative implications of new technologies (including CD-ROM and interactive laser disks), and desktop publishing (and desktop presenting) applications. These partnership explorations of new technologies help administrators recall the time needed to master each new innovation. That understanding will prove useful to them later when they plan and implement staff development projects designed to encourage the adoption of new technologies.

Each of the elements of the administrative computer competence course expectations are important. Without the hands-on experience, discussions become remote, second-hand experiences. Without the discussions, the hands-on experiences degenerate to software training workshops. Without the discussions and hands-on workshops, the project presentations become "Show-And -Tell" sessions. Together, the three components provide a solid foundation for an administrative computer leadership capable of guiding us steadily into the next decade. The course prepares administrators who can and will model appropriate technology uses in their own school classrooms and offices.

Langhorne, et al, voice a major concern of educational computing advocates: "Educators have an embarrassing history of adopting the latest fad, trying to make those fads meet all needs, becoming dissatisfied because the new idea did not bring success in every application, and so abandoning the innovation and moving on to the next." We don't want the computer to share the fate of instructional television and other educational innovations that came and went. Robert McCarthy expresses this concern well: "The computer is too powerful an educational tool to be cavalierly consigned to the educational broom closet."

If we are to keep the computer out of McCarthy's broom closet, we will need the vision and leadership of our school administrators. Our educational leaders must establish long range plans to guide their instructional computing programs. Daniel E. Kinnaman reports the recommendations of the National Education Association's committee on educational technology: "All schools should develop and implement a plan to install a computer with adequate software on the desk of each teacher by 1991. Resources for educational technology must be adequate to provide balanced support for staff development, software, hardware, and research into curriculum integration and development." To develop such plans we must have informed, knowledgeable, and dedicated planners—planners who have learned the powers and pitfalls of educational technology adoption. The only way to avoid the computer broom closet is to require our administrators to raise their computer competence. That task can best be accomplished with instruction dedicated to that purpose. A course like the one described here, or a similar one in a different format, should be a required as part of the preparation for receiving and maintaining administrative certification. Our educational leaders must develop a clear understanding of what they are expected to lead.

References footnoted in text

1. Barbour, Andrew. (April, 1987). "Office romance: Why administrators are hooked on technology." *Electronic Learning.*
2. Beaver, John F. (April, 1987). *A national search for quality: An examination of high-quality elementary school instructional computing programs.* Unpublished dissertation, Michigan State University.
3. Bruder, Isabelle. (June, 1989). "NSBA study says largest districts have weak long-range technology plans." *Electronic Learning.*
4. Charp, Sylvia. (June, 1989). Editorial. *Electronic Learning.*
5. Fulton, Kathleen. (October, 1988). "Preservice and Inservice: What Must Be Done in Both." *Electronic Learning.*
6. Kearsley, Greg (November, 1988). "What Should Today's Administrators Know About Computers?" *Technology Horizons in Higher Education Journal.*
7. Kinnaman, Daniel E. (September, 1989). "NEA committee on educational technology releases report." *Classroom Computer Learning.*
8. Langhorne, Mary Jo, Jean O. Donham, June F. Gross, Denise Rehmke. (1989). *Teaching with computers A new menu for the '90s.* Phoenix: Oryx
9. McCarthy, Robert. (September, 1988) "Making the future work: The road to computer integration." *Electronic Learning.*
10. Moursund, Dave. (June, 1989) "Why are our colleges of education continuing to graduate computer illiterate teachers?" *The Computing Teacher.*
11. Scrogan, Len. (January, 1989). "The OTA report: Teachers, trainers, and technology." *Classroom Computer Learning..*
12. Thornburg, David D. (October, 1988). "A time to take stock of the state of educational computing." *A+ Magazine..*

Paper
Selecting Personal Computers for a University MicroComputer Cluster: A Multiple Criteria Decision Analysis Approach

Philipp A. Djang
U.S. Army Training and Doctrine Analysis Command
White Sands Missile Range

Abstract

This paper describes a Multiple Criteria Decision Analysis Approach (MCDAA) to determine the selection of personal computers for a university personal computer cluster. The MCDAA combines the capabilities of Analytic Hierarchy Process (AHP) and Integer Goal Programming (IGP). Decision makers can use this approach to determine "what kind" of personal computers and "how many" of each type of personal computers could be purchased. A real world application of the MCDAA illustrates how the processes are combined to answer the "what kind" and "how many" questions.

Introduction

A university computer center is a dynamic organization that reflects the state of computing technology. The pressure to provide up-to-date educational computing facilities to the students, faculty and staff present a uniquely significant problem. As university computer centers adopt personal computers into their array of technology and develop the facilities and personnel staffing that are required to operate a personal computer cluster, two significant decisions confront them. These decision are to determine "what kind" of personal computers should be purchased and, given a limited budget, "how many" of each kind of personal computers can be purchased.

The purchase of a particular type of personal computer for a microcluster is based upon many criteria. The principal criterion for the selection of a personal computer in a university setting is based upon its educational value. Additional criteria are cost, maintenance, compatibility with current and future work environments, network compatibility and popularity with students. Since this is a significant capital outlay, it is important that a center select those computers that are consistent with the educational goals of the university.

A Multiple Criteria Decision Analysis Approach

Personal computer purchasing decisions can be expedited by an application of an integrated multiple criteria decision analysis approach. The integrated Multiple Criteria Decision Analysis Approach (MCDAA) combines the capabilities of Analytic Hierarchy Process (AHP) and Integer Goal Programming (IGP). The AHP can assist decision makers with the resolution of complex problems that involve multiple criteria [4]. Previously, the AHP has been used to select multi-micro distributed accounting information systems [1] and evaluate micro/mini computers configurations [2]. This paper extends these applications by describing a method for combining AHP and IGP so that the purchasing decision is founded upon both the subjective values of a decision maker and the objective constraints of money and space.

In this approach, decision makers identify, for each criteria, the relative preference for each of the decision alternatives by an application of the AHP. The result of the process is a prioritized ranking of each of the decision alternatives with respect to the perceptions of the decision maker. Each of the decision alternatives receives a ranked score on each of the criteria. The results quantify the decision maker's preference for each decision alternative and provide a means for answering the "type of computer" question. In order to determine "how many" personal computers to purchase, IGP can be used. IGP is a linear optimization method that incorporates the preferences of the decision maker [3]. The priority ranking for each type of alternative can be incorporated into an IGP. The AHP rankings provide an efficient means for determining the priority weights for the IGP.

The Analytic Hierarchy Process

The AHP is appropriate when the criteria cannot be accurately measured or the decision alternatives cannot be precisely differentiated. This particular methodology is used because one is basically interested in a value assessment of the deterministic factors rather than their utility [1]. The primary advantage of the AHP is that it provides a methodology that includes both objective, measurable criteria and the subjective perceptions of the decision makers. The methodology of the AHP requires that a hierarchical structure be developed for the decision. The hierarchical structure consists of three levels. The levels are the overall goal, the criteria and the decision alternatives.

The overall goal for this application will be to select the type of personal computers (decision alternatives) that satisfy the criteria. The decision alternatives for this application are the

IBM PS/2, an IBM compatible and the Apple MacIntosh. Additional brands of personal computers could be easily considered in the decision making process. The analytic hierarchy for a university microcluster is given in figure one.

For this application of the AHP, the criteria were divided into two classes, an educational class and a functional class. The educational class consisted of the criteria that could be classified as falling in the educational domain, while the functional class was composed of the criteria that dealt with daily operations. A total of fifteen criteria were generated. Five of the criteria fell into the educational class and the remaining ten criteria were classified as functional criteria. The decision maker was asked to compare the alternatives on each of the criteria.

The AHP methodology is based upon the construction and analysis of a pair-wise comparison matrix [4]. For each decision criteria, the decision maker compares two of the decision alternatives on the criteria and then ranks the comparison according to the scale of values (9=high preference, 1=equal preference). For each criteria, n(n-1)/2 comparisons were required. A positive matrix was constructed which consists of the values and their reciprocal of the values. The reciprocal values were entered into the transpose position of the square matrix. All decision criteria matrices are square, since all decision alternatives are compared. The diagonal of all the decision matrices are given the value of one (consistent with the rating scale). The pairwise comparison matrix is normalized by computing the sum of each row and dividing each element by the sum. A principal vector is constructed by averaging the normalized row values. For the criteria, the principal vector contains the relative priorities for each decision alternative.

The consistency of the rankings can be evaluated by a few matrix operations. The pairwise comparison matrix is multiplied by the principal vector. From the resulting matrix, the sum of the row means are divided by the principal vector. The result vector is summed and divided by the number of decision alternatives. The result, λ_{max}, is used in computing the consistency index (CI).

$$CI = \frac{\lambda_{max} - n}{n - 1} \qquad \text{(1) Consistency Index}$$

The consistency ratio (CR) is computed by dividing the CI from a value obtained from a table of random consistency. If the CR is less than 0.10, the comparison exhibits internal agreement. Otherwise, the criteria would require additional evaluation.

The Analytic Hierarchy Process produces a prioritized evaluation of each of the decision alternatives. Each of the decision alternatives (IBM, IBM compatible and Apple MacIntosh) receives a score on each of the criteria. The criteria scores are combined into an overall score. The overall score indicates the relative importance of each personal computer and assists the decision maker in determining where resources should be allocated. The results of the Analytic Hierarchy Process quantifies the decision maker's preferences for each decision alternative and provides a means of determining the "what type of computer to buy" question. In order to determine "how many" personal computers to purchase, integer goal programming can be used.

Goal: Select Microcluster Personal Computers

Educational Class	Functional Class
Student Preference	Purchase Costs Repair Cost
Homework Compatible	Space Limits Staff Skills
Ability to Operate Educational Software	Availability of Peripherals Compatible w/ Network Ease of Upgrades
Faculty Preference	Compatible w/ Software Vendor Relationship
Future Computing Environment of Employer	Vendor Market Standing

Decision Alternatives:
1) IBM PS/2
2) IBM Compatible
3) Apple MacIntosh

Figure 1. Analytic Hierarchy for a University Micro-Cluster

Integer Goal Programming

The priority ranking for each type of personal computer can be incorporated into a goal programming model. The AHP rankings provide an efficient means for determining the priority weights for a integer goal programming model. Goal programming is a linear programming method that is appropriate when an organization has multiple conflicting objectives and is attempting to satisfy all of the objectives under the constraints of scarce resources. Goal programming has the capabilities to analyze the decision maker's multiple aspiration levels, to relax some of the linear programming constraints and to incorporate the decision maker's preference system for multiple conflicting goals. Goal programming is a method for transforming an optimizing model into a satisficing model by incorporating the decision maker's preferences. [3] In goal programming, all of the objectives are assigned target levels (the goals) for achievement and a relative priority is placed upon achieving these levels. Goal programming attempts to find an optimal solution that comes as "close as possible" to the targets in the order of the specified priorities. [3]

A small integer goal programming model was developed to determine the quantity of personal computers that the university microcluster should purchase. The priorities for the goals were obtained from the AHP. A hypothetical funding level was used as a limiting constraint on the number of computers, network accessories and software could be purchased. Estimates of the actual physical space, personal computer, peripheral and network costs were obtained by interview with the Director of a university microcluster. The goal of the integer goal programming model is to satisfy the AHP priority values and at the same time, minimize the shortfall from the goal of purchasing an integer number of computers and associated peripherals, subject to funding and physical space constraints.

Pairwise Comparison Scale

Verbal Judgement of Preference	*Rating Scale*
extremely preferred	9
extreme to very strongly preferred	8
very strongly preferred	7
very strongly to strongly preferred	6
strongly preferred	5
strongly to moderately preferred	4
moderately preferred	3
moderately to equally preferred	2
equally preferred	1

Figure 2. Rating Scale

An Example

A structured interview/questionnaire was developed to elicit the priority of the criteria of the different computers. The director of a microcluster was interviewed and the rankings for each of the criteria were collected and analyzed according to the methods prescribed by the AHP. The results of applying the AHP produced a prioritized evaluation of each of the decision alternatives. The rating scale is given in figure two. For one of the criteria, the raw preferences (left side choice preferred over right) that the Director expressed were:

IBM PS/2 versus IBM Compatible = 3.0
IBM PS/2 versus MacIntosh = 3.0
MacIntosh versus IBM Compatible = 3.0

From these data, the square pairwise comparison matrix is constructed. The transpose elements of the matrix are the inverse of the original values. The sum of the columns is divided into the elements of the column to give the normalized matrix. The principal vector is the mean of each row. In order to compute the consistency of the priorities, the weighted sum and weighted mean are computed. λ_{max} is computed from the weighted means and used to calculate the consistency index. From a table of random consistency [4], for n = 3, the value of 0.58 is obtained. The consistency ratio computed from the tabled value and the consistency index. The pairwise matrix, principal vector, weighted mean, λ_{max} and consistency ratio is given in table one. The consistency rule indicates that this value is on the border of being acceptable. The rule states that values greater than 0.10 *may not* be consistent. However, the observed value has a small deviation from the rule and the raw data indicate that the decision maker's preference is linear. Based upon these observations, the judgments can be considered to be consistent.

A pairwise comparison of the remaining decision criteria are constructed in a similar manner. Each criteria is compared to each other criteria and a square comparison matrix is formed. This matrix is termed a *priority matrix*. The principal vector of the matrix contains the relative importance of each criteria. The next step is to apply the pairwise comparison method to the criteria. Each criteria is compared to each other criteria and given a value. The principal vector is computed and can be interpreted as the ranked scores for the decision criteria. Table two presents the criteria scores after the criteria have been compared to each other. These scores indicate the relative importance of each criteria.

Table 1
Pairwise Comparison Matrix

	IBM PS/2	*IBM Compt.*	*MacIntosh*	*Principal Vector*	*Weighted Mean*
IBM PS/2	1.0	3.0	0.3	0.28	3.13
IBM Compt.	0.3	1.0	0.3	0.14	3.05
MacIntosh	3.0	3.0	1.0	0.57	3.23

λ_{max} = 3.14 Consistency Ratio = 0.118

Table 2
Decision Criteria Pairwise Comparison

Educational Criteria	Row Mean	*Functional Criteria*	Row Mean		
Homework	0.337	Space	0.238	Repair Cost	0.062
Student Preference	0.257	Personnel Availability	0.152	Vendor Market Status	0.037
Future Job	0.182	Software Cost	0.152	Peripherals	0.037
Compatible w/ Software	0.127	Compatible w/ network	0.152	Procurement Cost	0.096
Faculty Preference	0.095	Upgrade Cost	0.037	Vendor Reliability	0.037

In order to construct an overall comparison for the decision alternatives from among the criterion rankings and the priority matrix, a final matrix operation is required. The element of a decision alternative in the principal vector of the priority matrix is multiplied by the corresponding element in each of the decision criteria's principal vector. The resulting sum is the overall priority weight for each decision alternative. Table three presents the scores for the educational, functional and overall scores. The overall score was computed by summing the educational and functional criteria scores.

The results indicate that the MacIntosh personal computer is preferred choice when educational criteria are considered. In contrast, for the functional criteria, the IBM PS/2 has been assigned the highest score. By combining the educational and functional criteria scores, it is observed that the order of preference is: the IBM PS/2, the MacIntosh and the IBM compatible. This application of Analytic Hierarchy Process demonstrates a method for comparing three different computers over multiple criteria.

Integer Goal Programming Formulation

In order to determine "how many" of each type of personal computers to purchased, an integer goal program was developed. The Analytic Hierarchy Process produced a prioritized evaluation of each of the decision alternatives with respect to the perceptions of the decision makers. Each of the decision alternatives (IBM, IBM compatible and Apple MacIntosh) received an overall score. The overall score indicates the relative order of importance for each personal computer and assists the decision makers in determining where the Microcluster should place its resources.

Table 3
Analytic Hierarchy Process Rankings

	Overall Priority	*Educational Priority*	*Functional Priority*
IBM PS/2	0.770	0.391	0.378
IBM Compatible	0.539	0.209	0.330
MacIntosh	0.690	0.399	0.291

The priority rankings that were observed from the application of the Analytic Hierarchy Process can be incorporated into a integer goal programming model. Goal programming is a linear programming method that is appropriate to use when the purpose is to satisfy multiple conflicting objectives under the constraints of scarce resources [3]. Goal programming has the capabilities to incorporate the decision maker's preference system for multiple conflicting goals. In goal programming, all of the objectives are assigned target levels (the goals) for achievement and a relative priority is placed upon achieving these levels. Goal programming will find an optimal solution by allocating resources to those objectives in the order of the specified priorities. [3]

Table 3 shows the competencies essential for inclusion in the database management portion of a single microcomputer applications course.

Table 3
Essential Database Competencies

Competency

Keyboarding
- Function keys
- Cursor movement keys
- Entering commands
- Entering records into a database
- Clearing the screen
- Creating a database file
- Creating the database file structure
- Defining the field names
- Defining the field types
- Defining the width and decimal positions

Editing
- Selecting a record
- Adding records to a file
- Inserting a record in a file
- Deleting records from a file
- Changing records in a file
- Modifying field contents
- Modifying record contents
- Deleting fields from the directory

Displaying
- Displaying selected records
- Displaying all fields
- Displaying selected fields
- Displaying a certain record
- Displaying using relational operators

Sorting
- Sorting records in descending or ascending sequence
- Sorting on more than one field

Printing
- Creating a report
- Designing the output
- Printing the report

File Management
- Displaying database files directory on disk storage
- Retrieving database files from diskette
- Deleting database files from diskette
- Making backup files of database files
- Copying a database file

Miscellaneous
- Help menu

Table 4 shows the competencies determined to be of some importance for inclusion in the word processing portion of a single microcomputer applications course.

Table 4
Important Word Processing Competencies

Competency

Keyboarding
- Status Line, Ruler Line, and Menu Line
- Goto Command
- Hard Spaces and Hard Returns
- Underlining Text
- Centering Headings
- Centering Lines
- Boldfacing
- Numbered Lists
- Creating Forms

Editing
- Undeleting Text
- Keyboarding Over Existing Text
- Searching for Words
- Searching and Replacing Text With or Without Confirm
- Backward Search
- Searching with Upper/Lower Case Capability
- Searching Using Wild Characters
- Searching Using Whole Word Capability
- Working with Blocks of Text
- Writing and Reading a Block of Text

Format
- Margin Release
- Justifying Margins
- Adjusting Margins
- Setting Lines/Inch
- Setting Characters/Inch
- Setting Tabs
- Tab Align
- Working With Indent Tabs
- Indenting Left and Right
- Setting Decimal Tabs
- Working With Decimal Tabs
- Adjusting Word Wrap
- Double Spacing
- Adjusting Line Spacing
- Working With Line Draw
- Centering Headings Over Columns
- Inserting Page Breaks
- Super/Subscript
- Centering Page Top to Bottom

Table 4
Important Word Processing Competencies
(continued)

File Management
- Storing Document Files to Disk With Overwrite
- Storing Document Files to Disk and Resuming
- Storing Document Files to Disk and Switching to Another File
- Changing Data Drive
- Abandoning a Document File
- Consolidating Two Document Files
- Locking or Protecting a File

Printing
- Printing Part of a Document
- Printing Current Page
- Print Screen
- Printing to Disk
- Printer Setup
- Conditional End of Page
- Printing More Than One Document
- Working With Different Printer Fonts and Printer Enhancements
- Printing With Hand Fed-Paper or Sheet Feeder
- Working With Printer Commands
- Working With Automatic Pagination
- Page Number Position Set-up
- Printing More Than One Copy
- Printing From Disk
- Printing Envelopes
- Printing Different Sized Paper

Miscellaneous
- Performing Spelling Functions
- Performing Thesaurus Functions
- Performing Grammar Functions
- Performing Style Functions
- Performing Math Functions
- Working With Columns and Column Functions
- Inserting Text Variables
- Boilerplating
- Composing Text
- Cancel Key
- Upper/Lower Case Convert
- Defining Macros
- Working With Text or Data Files
- Working With Windows
- Working With Orphans
- Working With More Than One Document File on the Screen
- Endnotes or Footnotes
- Merging Two or More Document Files
- Headers and Footers
- Working With Hyphenation
- Setting up Automatic Indexing
- Setting up Automatic Table of Contents
- Sorting Function

Table 5 shows the competencies determined to be of some importance for inclusion in the spreadsheet portion of a single microcomputer applications course.

Table 5
Important Spreadsheet Competencies

Competency

Keyboarding
- GoTo Command
- Entering Values with Decimal Default
- Formulas with Relative and/or Absolute References
- Using the Pointer to Enter Formulas

Editing
- Erasing (Blanking) a Rectangular Range
- Clearing the Entire Spreadsheet from Memory
- Revising by Striking over Characters
- Moving a Row
- Moving a Column
- Moving a Range

File Management
- Changing Data Drive
- Saving a Spreadsheet File on Disk with Overwrite
- Storing Document File to Disk and Resuming
- Storing Document File to Disk and Switching to Another File
- Displaying Spreadsheet Files on Disk Storage
- Consolidating Two Spreadsheet Files
- Loading or Saving Part of a Spreadsheet File
- Copying a Spreadsheet File
- Renaming a Spreadsheet File

Format
- Global Format
- Column or Row Format
- Adjusting Column Width
- Including the Dollar Sign with the Value
- Including Commas with the Value
- Designing Spreadsheet Layout
- User-Defined Format
- Column Total Lines and Underscores

Replication
- Copying a Formula into a Range
- Copying a Label into a Range of Cells
- Copying a Formula without Adjustment
- Filling In a Data Table

Functional Formulas
- Computing Counts with Blank Cells
- Finding the Largest Value
- Finding the Smallest Value
- Logical Functions

Table 5
Important Spreadsheet Competencies
(continued)

- Mathematical and Engineering/Scientific Functions
- Statistical and Financial Functions
- Calendar Functions
- Using IF Functions to Determine Status of a Value
- Using IF Functions to Make Computation(s)

Printing

- Printing a Report Without the Spreadsheet Borders
- Print Screen
- Printing to Disk
- Printer Setup

Miscellaneous

- Horizontal and/or Vertical Titles
- Horizontal or Vertical Windows
- Synchronized or Unsynchronized Split-Wise Scroll
- Building a Master Spreadsheet without Formulas
- Building a Master Spreadsheet with Formulas
- Protected/Unprotected Cell Entries
- Manual or Automatic Recalculation
- Defining Macros
- Sorting Function

Graphics

- Using Spreadsheet Data to Construct a Graph
- Specifying Data to be Graphed
- Defining Variables
- Formatting the Graph
- Viewing the Graph on the Screen
- Choosing Type of Graph
- Headings: Main, Sub, and Column
- Naming a Graph
- Saving a Graph
- Printing a Graph
- Creating Pie Charts
- Creating Line Graphs
- Creating Bar Graphs
- Creating Stacked Bar Graphs
- Creating Area Fill Graphs
- Creating XY Graphs
- Adding Labels to a Graph
- Changing the Graph
- Changing Graph Formats
- Creating Multiline/Multibar Graphs

Table 6 shows the competencies determined to be of some importance for inclusion in the database management portion of a single microcomputer applications course.

Table 6
Important Database Competencies

Competency

Editing

- Packing a File
- Browsing Through a File
- Combining Fields and Records from Two Database Files
- Correcting Commands
- Modifying the Database File Structure
- Updating a File With Data From Another File

Displaying

- Displaying All of the Records
- Displaying Next Record
- Displaying Using Logical Operators
- Displaying a Count of Records
- Displaying a Sum of a Field
- Displaying an Average of a Field
- Displaying a Sorted File

Sorting

- Sorting Selected Records From a File

Printing

- Preparing a Report Using Selected Records From a File
- Printing Reports With Subtotals
- Printing in Report Form
- Setting Left and Right Margins
- Setting Top and Bottom Margins

File Management

- Renaming Database File
- Changing Default Drive
- Protecting a Database File
- Entering and Running Command Files
- Editing and Correcting Command Files

Recommendations

Following a review of the findings of this study, the following recommendations can be made:

1. Software developers must strive to include all of the necessary microcomputer application competencies in their software and the materials that accompany their products.
2. Schools that are anticipating the addition of a course in microcomputer applications must be sure to include the essential competencies, as identified by this study, in this type of course.
3. Curriculum planners who are evaluating current course offerings, or content of courses currently offered, must verify that the essential competencies for a microcomputer applications course, as determined by the experts, be included in their curriculum.
4. Textbook authors should be sure to include the competencies identified by this study in the products they develop for a microcomputer applications course.

Projects

Computer Education Leadership and Graphic Art

Animation and Interactive Video in Multi Media Health Science Instruction

James Wall, John Strange
Department of Physical Therapy and Department of Education
University of South Alabama

Abstract

Interactive media offers enormous potential for the development of teaching materials for students in the health sciences. The major drawback with this technology has been the lack of appropriate programs, primarily because of the lack of programming expertise on the part of the faculty. Recently authoring systems which are extremely powerful and yet still relatively simple to use have been produced. As an example of the use of this technology, a number of interactive programs have been developed for use in the Physical Therapy curriculum. The computer screen of an Apple Macintosh is used to display textual material, graphics and animation while a second screen displays the output from a videodisk player. All the materials are interlinked and under the control of the student by way of the computer. A gait tutorial uses flip page type animations to show, for example, the spatial parameters, while more complex animations are used for showing such elements as the phasic activity of the muscles and the ground reaction forces. A videodisk is used to show actual walking sequences. A second program, being developed jointly with the University of Tennessee, Memphis, is designed to assist students review material on abnormal reflex development.

Guidelines for Better Screen Design

Penelope Semrau
California State University
Los Angeles, CA

Abstract

Many teachers today are creating their own educational software with authoring tools like *HyperCard*. Even though one has mastered the mechanics of *HyperCard*, making the software motivating is another matter. The integration of visuals in educational software creates a greater impact on the learner than the use of words alone. What is it that makes the screen aesthetic, motivating, and inviting? Software screens can be critiqued according to the following visual criteria:

- *Unity* is where each part of the screen is necessary.
- *Balance* creates visual equilibrium on the computer screen.
- *Style* is the combination of distinctive visual features of the screen design which give it a unifying character.
- *Dominance* is what appears as the strongest element on the screen.
- *Repetition* of shapes, patterns, or sizes of elements can unify a screen.
- *Contrast* is where opposites such as light and dark are placed closely together.
- *Arrangement* is the layout of the elements on the screen.

Using visual criteria in the design of software screens leads to professional-looking screens. The application of the visual design elements helps to better achieve educational objectives and promote quality learning experiences.

References

Apple Computer, Inc. (1989). *HyperCard stack design guidelines.* Reading, Mass.: Addison-Wesley Publishing Company, Inc.
Heinich, R., Molenda, M. & Russell, J. D. (1985). *Instructional media.* New York: John Wiley & Sons.
Hlynka, D. (1989). Educational technology: Bullet or butterfly? *AECT-RTD Newsletter*, 13(3), 7-10.
Semrau, P. (1990). *Checklist for good screen design.* Class handout for EDIT 484: Software Authoring Tools in Education. Los Angeles: California State University.

Preparing Computer Education Leadership for the 1990's and Beyond

Robert Hilgenfeld
Gonzaga University, Washington

Abstract

Abstract not available at press time.

Projects
Teacher Education Science Methods Planning Instruction

An Interactive Video Lesson Planning Project for Teacher Education

Charlotte Scherer
Bowling Green State University, Ohio

Abstract

This teacher education project involves the development of a hypercard stack to assist students in planning for instruction. The stack is designed to control the MECC Improving Teacher Effectiveness Laser Disc. It calls up appropriate visual segments as the student progresses through the process of creating a lesson plan, using many of the effective teaching strategies. Although effective teaching strategies are stressed, the lesson plan process in this program does not adhere to a single approach to planning. Rather, the Hypercard stack is designed to facilitate general lesson planning while allowing for incorporation of effective teaching strategies.

The materials will be pilot tested in the spring semester of 1991 with elementary education students enrolled in Computer Utilization in the Classroom courses who need to write lesson plans incorporating the computer as assignments for that class. Research will be done to test for the effectiveness of this lesson plan writing strategy by comparing the quality of lesson plans written using the entire IVD program with groups using just the hypercard stack and with groups writing lessons without the aid of any of the technology. The groups using the technology will also be asked for feedback regarding the use of the media itself. These teacher education students already have an assignment to use the Improving Teacher Effectiveness Laser Disc and Apple IIe programm designed by MECC. In this project they analyze and apply principles that promote student learning. The new Planning for Instruction segment using Hypercard will be a followup to the earlier assignment and will be done at the time in the semester appropriate to the lesson plan assignment.

Both the development process of the IVD program and the preliminary results of the research will be discussed at this session.

One Banana, Two Banana—Science and Math with Integrated Applications

Elizabeth Viau
Montana State University

Abstract

Integrated applications, such as Appleworks or MicroSoft Works, can be taught in the context of scientific investigation. One such project, used with pre-service math and science teachers, uses the scientific method , graphing, and statistics (if students are familiar with statistical analysis) to investigate bunches of bananas at local food stores.

Students develop hypotheses about how many bananas will be in a bunch. Then they go to the store and count inner and outer bananas in a specified number of bunches. Students record the time of sampling and the store where the sample is recorded. They return to class to enter their findings in a data base, and individual data bases are merged. The data is sorted, and summaries are put into the spreadsheet, averaged, and graphed. The students complete the assignment by writing up the experiment in the word processor, including information transferred from the data base and spread sheet. They are required to come up with tentative explanations for their findings, for example, for why banana bunches do not have the same means at different times of the day, and why the means differ from store to store. The ratio of inner to outer bananas can be graphed, and plotting bunch size versus frequency of bunch occurrence yields a normal curve.

A project of this nature allows students to practice the scientific method and the use of an integrated application with materials that are concrete and widely available. The project gives them an opportunity to work with raw data and practice the higher level, problem-solving skills involved in doing science.

Multimedia in Teacher Education

Ronald J. Abate
Cleveland State University, Ohio

Abstract

With funding from the Ohio Board of Regents, the Cleveland State University's College of Education is implementing multimedia instructional materials in its core teaching curriculum. One goal of the multimedia project is to alter the role of instructor from that of lecturer to that of mentor. A second goal is to provide a vehicle for increased student involvement. To achieve these goals four types of multimedia applications have been developed. They include computer controlled videodisc presentations, simulations, teacher tools and a hypermedia database.

- Presentations: The use of videodisc presentations in the classroom has improved student participation by providing undergraduates with shared observational experiences while the addition of computer control over video presentations has increased instructional flexibility. As an indirect result, the videodisc materials have provided a practical reason for coordinating student assignments across courses. For example, classroom management principles observed on videodiscs provided in a reading methods course are then applied in lessons plans created by students in a general methods course.
- Simulations: The courses targeted by the project have an extensive field component. In order to help students with their off campus experiences, tutorials and simulations were developed. These stand alone instructional applications provide an opportunity to practice specific skills prior to attempting these skills in the field. These applications have proven especially appealing to students with little prior experience in a classroom setting.
- Teacher Tools: Computer programs for the production of slides,video, audio and graphics have been introduced and workshops conducted to assist both faculty and students in the creation of instructional materials. Increasingly the focus of the project has moved towards integrating the faculty into the development process.
- Hypermedia: An ongoing effort in the project is the addition of exemplary lessons to a multimedia database. To date, eight lessons have been added to the database with additional lessons presently in development. Each lesson comes complete with text based background information, references, still frame images of student work as well as a video which may be viewed in a linear fashion or accessed via connections in the database. This application has proven especially interesting to our students. It offers a link between what is described in the textbooks and what occurs in the classroom. Although assignments are used to engage the students with the lessons, the database is available to browse through in an open ended manner.

During the past year, this project has enjoyed increased acceptance by faculty who teach the core curriculum. Experiences on how and why the faculty have chosen to implement these innovative materials in their courses will be shared.

Exciting Projects for Native American Students

A Multimedia Approach to Teaching the Ojibwe Language

Richard Sgarlotti
Hannahville Indian School
Wilson, Michigan

Abstract

The Hannahville Indian School is the only K-12 Indian school in Michigan. It is located on the Potawatomi reservation near Escanaba. Although the tribe has decided that it is important to preserve the Ojibwe language within the community, there are few elders who still speak the language, and only a few students who have seemed interested. However, the students are very interested in computer technology.

The school has been selected as one of the Apple Crossroads Grant recipients, and is using the preservation of the language as one of the focal areas for development of new programs.

This will include the use of video disk technology, with the English language on one audio track, and Ojibwe on the other. It will also involve the use of a Macrecorder, not only to record elders and other native speakers, but also to enable students to record their voices and compare them with the elders'. The audio and video presentations will not only focus on language, but also the culture, including art, music, and history. All of this will be driven by a HyperCard stack, which will also contain scanned images.

The preparation of this will be a cooperative, whole language effort of students and teachers in all subject areas, and community members.

Recreating Native American Legends with HyperCard

George Beekman
Computer Science Department
Oregon State University, Corvallis

Abstract

For the past two summers Oregon State University has hosted math camps for the American Indian Science and Engineering Society (AISES). The OSU/AISES Math Camp provided junior high Native American students with learning experiences in mathematics, computers, and communication. In addition, the camp provided activities designed to help them better understand and assimilate their Indian heritage.

Students had about twenty hours of hands-on computer class time during the two-week camp. In the first week they learn the basics of working with Macintosh computers and HyperCard. During the second week students worked on group projects.

The resulting projects are interactive graphic recreations of traditional Native American legends. Many are animated; some contain sound effects and dialog. In general, the projects look like they were created by seasoned computer users rather that by junior high students with little or no computer experience.

HyperCard's visual authoring environment made it possible for students to quickly learn the fundamentals of computer operation, text editing, graphics creation, and program design while creating impressive and culturally significant projects. The Math Camp experience suggest that tools like HyperCard have potential to transform introductory computer education.

The Native American Educational Network: Its Impact on Teachers and Administrators in Geogrpaphically Isolated Schools

David Basham
Northern Arizona State University

Abstract

The Native American Educational Network (ENAN) originated in 1988 as the Easter Navajo Agency Network. Funded by a grant from the Tandy Corporation, the Bureau of Indian Affairs and the University of New Mexico, ENAN initially served six BIA schools on the Navajo Reservation in western New Mexico. By the fall of 1989, ENAN had linked up as many as 15 schools outside of New Mexico.

Over a three-day period, nearly 70 callers from over 36 locations utilized the network. This kind of network traffic is not remarkable unless one knows where they originated. Lukachukai, Nazlini, Dennehotso, Chinle, Red Mesa, Wingate, Canoncito and Crownpoint are all very remote communities on or near the Navajo Reservation in Arizona or New Mexico. South Dakota is well represented, with callers from Vermillion, Cheyenne River, Agency Village, Marty Indian School and Aberdeen. Networkers also include teachers and students from Idaho, from the Zuni Reservation in New Mexico, Lawrence and Overland Park in Kansas, Muskogee, Oklahoma, and several BIA offices, as well as BIA officials from Washington, DC.

Students and teachers at the University of New Mexico, Navajo Community College, Northern Arizona University and Haskell College also utilize ENAN. Several university courses are offered on the network, but the bulk of usage involves collegial exchange among teachers, student projectsat the rural schools, and administrative coordination among the BIA schools and their Agency administrators. Most readers will not find many of these rural Native American schools on their maps, but for the first time these isolated teachers and students have the means to create and explore the new learning environment provided by computer-mediated communication.

Paper

HyperStories: Cultivating Interactive Reading and Writing on the Macintosh Computer

Joyce Brody and Mary Lange
San Dieso Unified School District, California

Abstract

HyperStories utilizes *HyperCard*, a Macintosh computer application software program, to inspire students to become actively involved with the reading and writing process. The students interact, on the computer, with both stories they create, and the stories of other students.

HyperStories begins with one story line or Story Starter, then branches off into various continuations of the original story. It is a very open-ended program, which has been automated on the Macintosh computer. It is a non-linear program, that allows each reader and writer to make her/his own decisions for stories that branch.The students must incorporate problem solving strategies in both the designing (writing) and utilizing (reading) of a *HyperStory*. Little or no computer programming skills are required from either the student or the teacher to use the *HyperStories* program.

We have been researching and developing applications for the *HyperStories* Project in the San Diego City Schools Deaf and Hard of Hearing program from September of 1988 through June of 1991. The students that have been involved in this project have been between the ages of seven and twenty two years old. Their skill levels have ranged from students who have just recently begun learning the English language, to those students who have been fully mainstreamed in regular high school English classes.

Teachers can easily adapt *HyperStories* to almost any type of classroom, curriculum or student. The students enjoy working and utilizing the writing process via a new medium. *HyperStories* offers a novel and exciting way for students to explore their creativity, while focusing on the writing process.

HyperStories was originally developed to make reading and writing enjoyable for elementary through high school age Deaf and Hard of Hearing students. Our goals are to concurrently motivate and improve the skills of our students in the domains of reading and writing. This program is excellent for promoting reading and writing, not only for the deaf, but for all students! Our experience has shown *HyperStories* to be successful with students who have shown the least enthusiasm for traditional presentations of the reading and writing process.

HyperStories: The Concept

HyperStories utilizes *HyperCard*, a Macintosh computer application software program, to inspire students to become actively involved with their stories. The students interact, on the Macintosh computer, with both their own stories and the stories of other students. *HyperCard* is a software program that allows the user to devise his or her own path for reading and writing. It is a non-linear program, in which the learners make their own decisions. Each experience with the program is distinct. *HyperStories* is an open-ended program, which has automated this non-linear process on the Macintosh computer, and it requires little or no computer programming skills from either the student or the teacher.

We selected the *HyperCard* program for *HyperStories*, because it is a relatively easy program to learn, and it has the capabilities of merging text and graphics effortlessly on a single page. The text, graphics and linking capacities of *HyperCard* enable students to develop and illustrate not only their own stories, but to choose their own sequence for reading and writing the stories. Students can draw their own pictures, utilize clip art, or scan in photographs to illustrate their stories. This experience most nearly duplicates the creation and publication of a real book. Students feel a real sense of accomplishment and pride at the completion and printing of their own *HyperStories*.

HyperStories begins with one story line, or Story Starter. It then branches off into various continuations of the original story. One student originates the introduction of the setting, main characters, theme and plot. Then at some exciting point, the student envisions two or three possible corollaries for the story. S/he passes on the story to another classmate, a student at another school, or perhaps s/he sends the story across the country. Different students then continue the story's various branches, until the story reaches another appropriate point for branching. In this way, the story proceeds to its disparate conclusions.

We decided to develop a schedule to assist the various classes to integrate the writing of *HyperStories*. A Story Starter introduces the story; establishing a setting, characters, theme, and the beginning of a plot. Then at some dramatic point in the story, the student creates three choices or story branches. The story is then passed on to another person or class. As the stories continue, we limit the number of choices, or story branches, and schedule a conclusion to another branch of the story, to keep the stories from becoming unwieldy. The schedule in Figure 1 has proved very successful for our program.

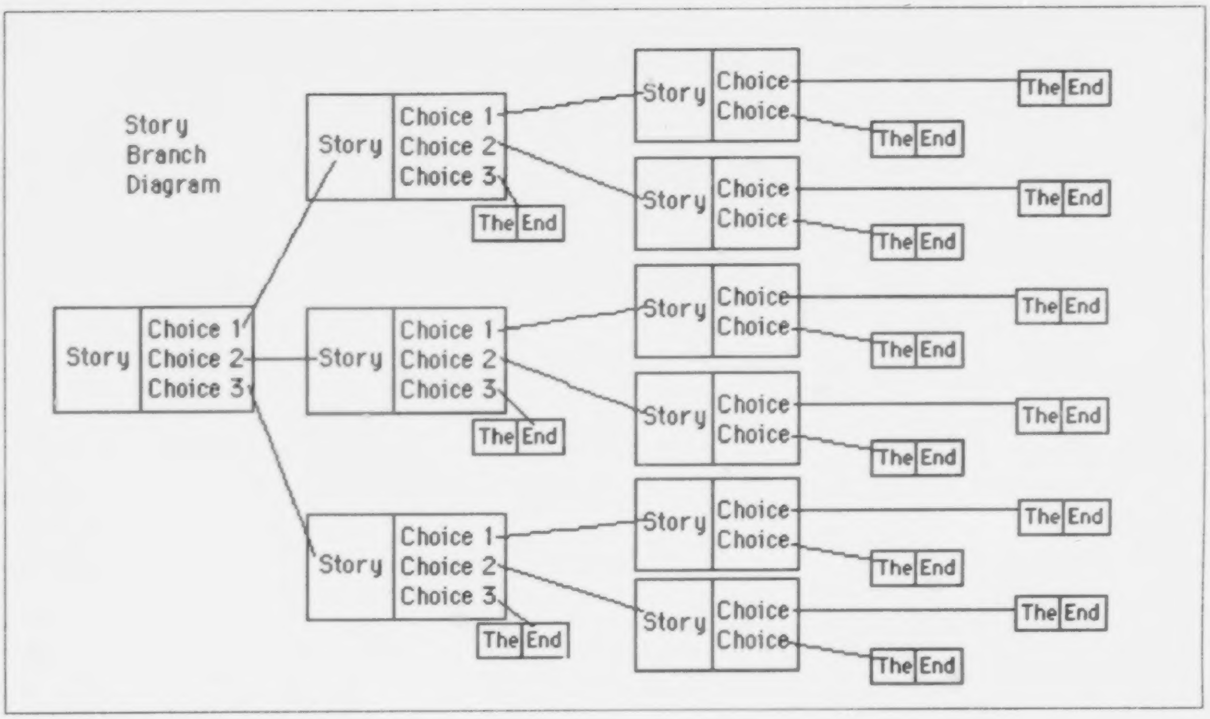
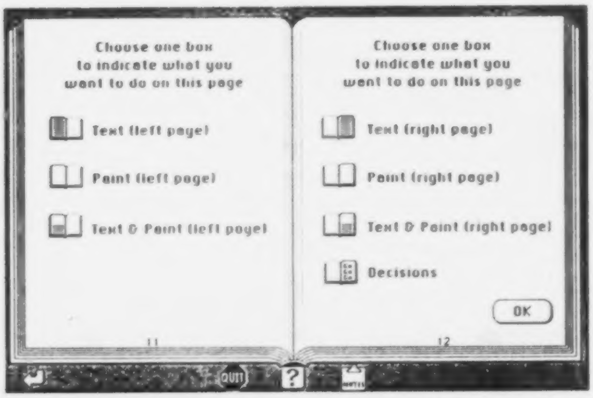

Story
Branch
Diagram
Story
Choice 1
Choice 2
Choice 3
Story
Choice 1
Choice 2
Choice 3
The End
Story
Choice
Choice
The End
The End

Figure 1. Sample story branch schedule.

***HyperStories* : Create Your Own**

To create a *HyperStory*, a student is constantly involved in a decision-making process. Not only is the student engrossed with the traditional decisions involved with writing an original story, but also with the decisions of the layout for each page of the published product. Decisions are somewhat limited, by the programming of the software, to seven basic choices and any combination thereof. Students can select a pre-programmed page layout by clicking one icon on each page and then clicking **OK**:



Choose one box to indicate what you want to do on this page

Choose one box to indicate what you want to do on this page

Text (left page)
Paint (left page)
Text & Paint (left page)
Text (right page)
Paint (right page)
Text & Paint (right page)
Decisions
OK
11
12
QUIT
?

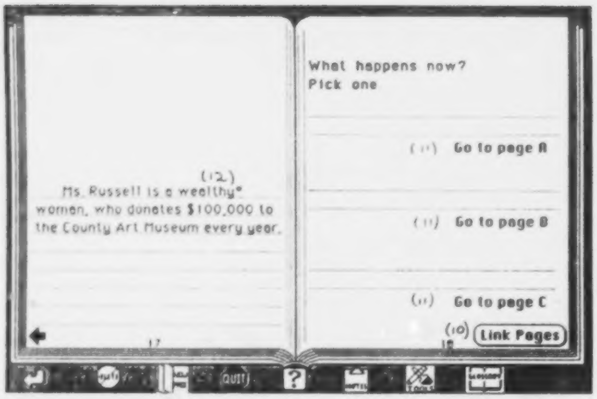
Figure 2. Sample *HyperStories* page layout selection.

The sample on the page below demonstrates two *HyperStory* pages, which have already been selected. One page is set up for Text and Paint (left page). Here a student can draw, scan or utilize clip art to illustrate a story and continue writing the story directly under the graphic. The right page is set up for Decisions (three). A student can envision three possible paths upon which her/his story will be continued by another student (Figure 3.)

Buttons, primarily represented by icons, are located at the bottom of each *HyperStories* page (See Figure 3). Students can click on the buttons and do the following:

Numbers on Figure 3, correspond to the explanations below.

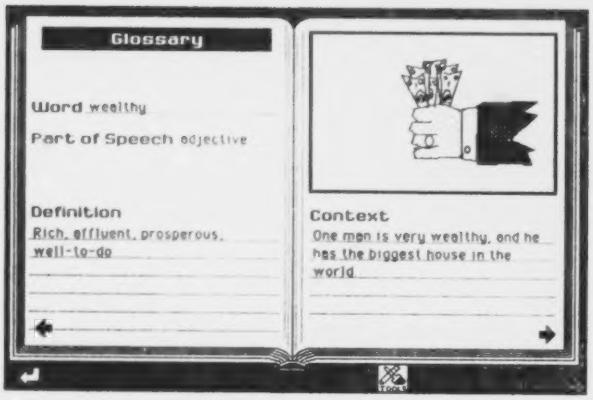
(1) Turns back to previous page.
(2) Turns back to previous story branch.
(3) Deletes the card and returns to the previous page. It will only appear when it is possible to delete a card.
(4) Makes a new choice page. This button also will only appear when it is possible to add a new page.
(5) Closes *HyperStories* and quit *HyperCard*.
(6) Takes you to a credit and teacher options page.
(7) Takes you to a "story ideas" page that gives students a place to organize or summarize their ideas for their stories.
(8) Shows the tool window. You must select the browse tool before closing the tool window.



Ms. Russell is a wealthy* women, who donates $100,000 to the County Art Museum every year.

What happens now?
Pick one

Figure 3. *HyperStories* page and explanation of buttons.



Glossary

Word wealthy

Part of Speech adjective

Definition
Rich, affluent, prosperous, well-to-do

Context
One man is very wealthy, and he has the biggest house in the world

Figure 4. Sample *HyperStories* glossary page.

(9) Makes a new glossary page, and adds a new word to the glossary, or looks up a word already in the glossary.
(10) Automatically links the number of decisions the student has selected to the appropriate page.
(11) After linking the pages for a decision, the Go to page A, Go to page B, and Go to page C will become buttons that bring you to actual sequential pages.
(12) Asterisked vocabulary word may be entered or "looked up" in the glossary.

The glossary button is a distinctive button in *HyperStories*. It serves a two-fold purpose. It enables students to instantaneously "look up" unknown, asterisked words in the glossary, by highlighting the words and then clicking on the glossary button at the bottom of the page. Similarly, students, at any time, can add their own words to the glossary by highlighting one or more words and clicking on the glossary button at the bottom of the page. Options on the glossary page, include the word's; part of speech, definition, and a sentence with the vocabulary word used in a new context. A space is also provided for students to draw a picture to better clarify the meaning of the word where it is appropriate (see Figure 4).

HyperStories: The Project

The development of *HyperStories* was funded by an Apple Equal Time Education Affairs Grant for the academic years of 1988-90. The Apple grant was awarded to selected applicants who addressed the needs of one of the following groups of students in educational settings; economically disadvantaged, ethnic and linguistic minorities, physically and mentally disabled, or females in math and science.

Our project met the grant's requirements of serving the linguistic minority and physically disabled students of the Deaf and Hard of Hearing program in the San Diego City Schools. The grant was written in collaboration with the Educational Technology Department of San Diego State University (SDSU) and Professor Bernie Dodge. Some of the graduate students from SDSU Educational Technology Department assisted in the development of tutorial stacks to supplement *HyperStories*, Story Starters (the introduction of a story line and the first three possible story branches), and completed *HyperStories* to be used as both reading materials and examples for students.

We researched and developed applications of the *HyperStories* Project in the San Diego City Schools Deaf and Hard of Hearing program from September of 1988 through July of 1990. Approximately 83 Deaf and Hard of Hearing students, in 12 classes, from the second grade through high school have been involved in this project.

HyperStories is versatile, and it lends itself to modifications according to the needs of both the students and the teachers. Students have ranged in ability from those who just recently began learning the English language as well as reading, writing and sign language, to those students who have been fully mainstreamed in regular high school English classes.

In the development of *HyperStories*, we have had particular target areas for improving the reading and writing skills of our students. We have focused on the development of consistent characterization, setting, plot, theme, cohesion and fluency in the writing domain. We also focused on the same facets of reading, while special attention was paid to comprehension skills. It is extremely difficult for a person to successfully continue a story someone else has written, without fully understanding the previous portion of the story. We also have emphasized the need of students to increase both their receptive (reading comprehension) and expressive (written) vocabulary through the use of a glossary page and context clues.

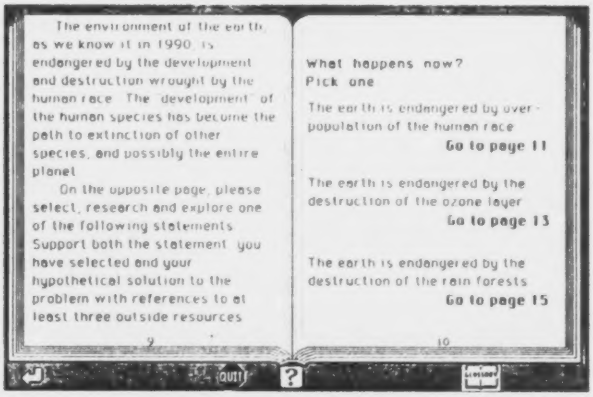
The Process

The first year of the *HyperStories* project was primarily utilized to introduce the Macintosh SE computers, *HyperCard*, and *HyperStories* to the staff and students. It was also used to ascertain their skills in the use of *HyperStories*. We found that the students were far more flexible and open to the use of new technology in the form of *HyperCard* and the Macintosh computers than were the teachers. Many of the teachers were originally dubious about their students' and their own abilities to learn to apply *HyperStories* in an expedient manner. They were extremely concerned that the process (using the Macintosh computer and *HyperCard*) not preempt our true goals of teaching reading and writing. It was also necessary to assist teachers in finding ways to incorporate *HyperStories* into their district mandated curriculum.

We devised a method of introducing the graphics exponent of *HyperCard* to the entire Deaf and Hard of Hearing program that became very popular. In October, all the students drew haunted houses. They used the linking capabilities of *HyperCard* to "open" and "close" doors, windows, and chimneys (by clicking on transparent buttons) to enable an audience to "look inside" and see the witches, ghosts, goblins, and so on, linked by the buttons. The students were thus introduced not only to *HyperCard* and the use of its graphics tools, but they were also surreptitiously initiated into the concept of linking pages through *HyperCard*.

Once the students became comfortable with the graphics, we had them write a linear story using the *HyperStory* Template as a pre-test. Next, students began working within their classes with the story-branching concept. They were then ready to write a *HyperStory*.

Teachers easily adapted *HyperStories* to almost any type of classroom, curriculum or student. It offered a novel and exciting way for students to explore new ideas. In second grade, some students merely drew pictures and either dictated or typed the words to their story. In high school, some teachers utilized *HyperStories* to promote creativity and problem solving techniques. The example below, demonstrates a possible use of *HyperStories* in a high school science class:



The environment of the earth as we know it in 1990, is endangered by the development and destruction wrought by the human race. The "development" of the human species has become the path to extinction of other species, and possibly the entire planet.

On the opposite page, please select, research and explore one of the following statements. Support both the statement you have selected and your hypothetical solution to the problem with references to at least three outside resources.

What happens now?
Pick one

The earth is endangered by over-population of the human race
Go to page 11

The earth is endangered by the destruction of the ozone layer
Go to page 13

The earth is endangered by the destruction of the rain forests
Go to page 15

Figure 5. Sample *HyperStories* integration into curriculum.

In the elementary classes, students integrated dictionary skills in completing the glossary pages, and in the junior and senior high, the glossary pages provided students with an inducement for using a thesaurus. High school students became very excited about taking an active part in building their own expressive and receptive vocabulary. They were able to choose the words they were interested in learning, from their own creative stories, rather than depending on a teacher to prepare lists of vocabulary words for them to memorize. An interesting ancillary consequence of having the students read and continue each other's stories, was an added incentive to learning new vocabulary words. The students did not want their fellow high school students, particularly younger ones, to know longer vocabulary words than they did.

The Results

In October of 1988, as a precursor to the project, students were asked to write a linear story. Students at the elementary level were shown a variety of well-known books as stimuli. In the junior and senior high schools, the students were simply asked to write a story. Eighteen months later, in March of 1990, the students were asked to repeat this task as a post test.

Extensive research revealed no existing assessments of story development in the specific areas of character, setting and/or plot development. Mary Lange, the second author, designed a three part evaluation for this project.

Students were rated on three different aspects of their stories. The assessments include: 1) word count 2) existence of a story 3) quality of story.

Simply counting the quantity of words in a written language sample is a typical method of assessing the writing of Deaf and Hard of Hearing students. The number of words correlates with the number of propositions. (Yoshinaga, 1983) Stories were also analyzed for story components loosely based on the Stein & Glenn (1979) story grammar analysis.

We predicted that the Deaf and Hard of Hearing students' stories would improve, but minimally through learning and utilizing *HyperStories*. We merely hoped to find a new way of inspiring students to become involved in reading and writing. Based on the evaluations described above, when the scores of pre and post writing samples were compared, there was an average score increase of 73% in overall story development! There was no correlation between student scores and their level of hearing loss. The average increase in word count was 67%. In the pretest, four of the writing samples did not qualify as stories, whereas in the post-test, only one of these was unable to qualify. An average increase of 78% was found in the quality of the stories. (Lange and Dodge, 1991)

The students were surveyed in June of 1990 on their opinions of the Macintosh computer compared to the Apple *IIe*. Results proved that they all enjoyed using both the Apple *IIe* and Macintosh computers; however most students chose the Macintosh over the Apple *IIe* computer. Students chose the Macintosh computer as their preferred tool for reading, writing, and drawing over both the Apple *IIe* computer and paper/pencil/book by at least a two to one margin.

Classroom teachers were highly enthusiastic about the *HyperStories* Project. They noted great improvements in both the quantity and quality of student-written language samples. They noticed the largest improvements in areas of clarity and cohesiveness. They also commented on the motivation students began to exhibit for both reading and writing.

HyperStories : Supplemental Ideas

The *HyperStories* Template can be readily adapted to a variety of age levels, teaching styles, and academic areas. It is basically a "shell" through which students and/or teachers can create. There are two other possible variations we have used with the template. In addition to stories which branch, students can also utilize the template to create linear stories. A simulation could also be constructed, taking advantage of the branching and graphics capabilities of the program. For example, an object could be hidden in a house. The student would need to "move" as s/he searches; in the living room, in the kitchen, and so on to find the object.

Accompanying the *HyperStories* Template are various *HyperCard* support stacks created by graduate students in the Educational Technology Department of SDSU. Character and Plot was developed by Carol Burnett and Gabriel Acosta. Cause and Effect was created by Gabriel Acosta and Joyce Brody. There have also been disks of Story Starters and completed *HyperStories* to promote *HyperStories*. These disks are available through: CUE SoftSwap, (Computer Using Educators).

Conclusion

HyperStories has proved very successful in motivating students to become involved in both reading and writing. Students not only enjoy reading and writing on the Macintosh, but they improve their skills. The writing process is utilized, vocabulary is developed and self esteem is enhanced through the publication of student created projects. Students most "turned off" by traditional reading and writing curriculums flourish using *HyperStories*.

References

Applebee, A.N. (1978). *The child's concept of story: Ages two to seventeen.* Chicago: University of Chicago Press.

Lange, Mary (1990). *HyperStories Report*. Final report submitted to The Apple Equal Time Grant Foundation. Apple Computer, Inc., Cupertino, CA.

Lange, Mary , Dodge, Bernie (1991). *HyperStories: An Interactive Story Writing Environment* , (Unpublished article, in process.)

Scally, Kevin (1984). *The Magic Road: The Story of Red Riding Hood*, London: Mushroom Books Ltd., Grosset &Dunlap.

Stein, N. L. & C. G. Glenn (1979). An analysis of story comprehension in elementary school children. In R. Freddle (ed.) *New directions in discourse processing.*, Norwood, NJ: Abex. [ED 121474]

Story Tree , Scholastic, Inc. P. O. Box 7502, Jefferson City, MO. 65102.

Yoshinaga, C. (January, 1983). *Syntactic and semantic Characteristics in the Written Language of Hearing Impaired and Normally Hearing School-Aged Children.* 206 p. PhD Dissertation, Northwestern University, Evanston, IL. [ED239467 EC161805].

Paper
Managing Multimedia Applications Using DESQview

David L. Doss
Bill E. Swafford
Illinois State University

Abstract

Current hardware technology provides functional microcomputer control of multimedia devices such as CD ROM, LaserDisc, and videoplayers. However, the limitation in current microcomputer operating system technology ofsupporting only a single executing task at a time, prevents full access to these resources. This paper describes how a commercially available windowing system, DESQview, can be used to extend the capabilities of DOS to allow access to these resources through a lesson management program written in an authoring language such as PC Pilot.

Introduction

Currently available technology provides highly functional control of multimedia devices for educational applications. Such devices as CD ROM players can be attached directly to SCSI hard disk controllers, while LaserDisc players can be accessed through a serial communication port. Currently, video players require a specialized interface, but new versions will soon be available that can be controlled through a serial port like LaserDisc players. These devices are typically controlled by issuing commands that establish position on the specific media, followed by a command to begin playing and for how long. The method of issuing commands varies from sending them directly to an output port to which a device is attached with some simple programming language like BASIC, to making requests to device drivers supplied with the hardware or the media through an authoring language such as PC Pilot [Rhodes & Azbell].

Each of these types of media have clear applications to education [Rhodes]. CD ROM players provide access to electronic encyclopedias and other information data bases. LaserDisc players give access to libraries of information and graphics images. Video players allow the display of commercially available video tapes or special tapes recorded locally. The general idea in applying this technology is to use an authoring language to manage the resources and carry on an interactive conversation with a student. Programs written in an authoring language can present options and record student responses.

This paper describes a method whereby all three media types can be combined in a single application using DESQview, a commercial software windowing package. This program supports the execution of multiple DOS programs simultaneously on a single PC. Using this feature, programs that control the video output of the LaserDisc or VCR, can be run concurrently with programs that generate output from the CD ROM player or continue to converse with a student who is using the system. These programs can be active in separate DESQview windows, with a user selected foreground window in charge of the PC display and keyboard. Any window can be selected as the foreground window, at any time.

Equipment Requirements

The DESQview software will work with virtually any true IBM PC compatible microcomputer. However, to obtain an acceptable level of performance in managing the simultaneous execution of several sizeable programs, a hard disk is required. The hard disk can be used to temporarily save unused program storage areas so that the memory occupied by them can be used by other programs. A much better solution is to use an AT class microcomputer with two to three megabytes of extended memory. Accessing extended memory is measured in nanoseconds as opposed to milliseconds for hard disks. Thus swapping programs in and out of extended memory is several orders of magnitude faster than the equivalent operation using a hard disk.

The system monitor should be able to display both text and graphics. This is not required by the DESQview software, but rather by the intended application. The monitor will be used to display the information from the CD ROM player using such facilities as an encyclopedia, which is highly likely to contain graphics images as well as text. The minimal CGA adapter will suffice for many current applications, but the newest software seems to use enhanced graphics. This is a good time to look to the future since EGA and VGA adapters provide CGA emulation modes, and thus will serve well now and for some time to come.

The LaserDisc and VCR will require a video monitor that is separate from the microcomputer console display since they both produce video and audio signals. Both devices can be attached to a single monitor because only one device is in use at a time. Both devices must also be attached to the microcomputer since they will receive control signals from it. The LaserDisc attaches directly to the serial communication port so that no special adapter is required. In the case of the VCR, a

```
1100 '            DISC Operations
1110 OPEN "COM1:4800,N,8,1,CS0,DS0,CD0" FOR RANDOM AS #1
1120 CLS : AA$ = "SA": PRINT #1, AA$: GOSUB 1400
1125  '
1130  '           Get user choice
1133  '
1135 CLS : LOCATE 6, 9: PRINT "VIDEO CONTROL MENU"
1138 LOCATE 9,  10: PRINT "1  Show Single Frame"
1140 LOCATE 10, 10: PRINT "2  Play Video Segment"
1145 LOCATE 11, 10: PRINT "3  Go To Next Frame  "
1150 LOCATE 12, 10: PRINT "4  Pause On          "
1153 LOCATE 13, 10: PRINT "5  Pause Off         "
1155 LOCATE 14, 10: PRINT "6  Stop              "
1160 LOCATE 15, 10: PRINT "7  Fast Forward"
1165 LOCATE 16, 10: PRINT "8  Reset"
1170 LOCATE 17, 10: PRINT "9  Quit"
1175 LOCATE 19, 2: PRINT "Enter number of desired activity -"
1176 IN$ = INKEY$: IF IN$ = "" GOTO 1176
1177 IF IN$ < "1" OR IN$ > "9" THEN GOTO 1135
1178 CHOICE = VAL(IN$)
1180 ON CHOICE GOTO 1505,1600,1610,1620,1625,1630,1640,1630,1660
1183 GOTO 1135
1185  '
1190 PRINT #1, AA$
1200 GOTO 1135
1400 '      SUB TO READ DISC PLAYER
1403 FOR I = 1 TO 200: NEXT     ' TIMER DELAY
1405 INPUT #1, A$
1410RETURN
1500 '      CHOICES
1505 GOSUB 620: AA$ = ARG2$ + "SE": GOTO 1190
1600 GOSUB 660: AA$ = "FR"+ARG1$+"SE "+ARG2$+"PL": GOTO 1190
1610 AA$ = "SF": GOTO 1190
1620 AA$ = "ST": GOTO 1190
1625 AA$ = "PL": GOTO 1190
1630 AA$ = "PA": GOTO 1190
1640 AA$ = "NFNFNFNF": GOTO 1190
1660 AA$ = "RJ": PRINT #1, AA$: GOSUB 1405: CLOSE #1: GOTO 90
```

Figure 1. BASIC Control of LaserDisc

special, and expensive, adapter card must be installed in the PC. It is also the case that the VCR must have two channel capacity, one for addressing and one for video signal. The CD ROM attaches to the PC with a SCSI interface adapter, which is a high performance microcomputer hard disk interface adapter, and therefore a standard commodity.

Software Requirements

In addition to programs supplied with the hardware, such as device drivers, and the DESQview system, three types of software are required: locally developed device control programs, locally developed lesson management programs, and commercial media software platters for the CD ROM and LaserDisc players. CD ROM platters typically come with control programs that provide all access to the data contained on them. LaserDisc platters currently come with a directory that provides addressing information for the data contained on them, necessitating some programming to obtain any desired information. Similar programming is required to establish position and initiate video player output. Such programs can be written in a simple language like BASIC or in an authoring language such as PC Pilot. Incomplete but informative program segments are shown for BASIC control of the LaserDisc in Figure 1 and PC Pilot control of the video player in Figure 2. Complete working programs are available upon request.

As can be seen in these two examples, both the device control and lesson management programming can be performed in BASIC or an authoring language. The segment in PC Pilot looks formidable to the uninitiated, but is actually relatively simple to learn. The advantage in using an authoring language is that much of the tedious detail of screen positioning and response recording is eliminated or at least significantly reduced.

DESQview

DESQview is a windowing system that enhances the functionality of the DOS operating system by providing support for a limited but powerful form of multitasking. The technique of time slicing is used to share the resources of the microcomputer between a single program in the foreground window and multiple programs in background windows. The program in the foreground window controls the keyboard and the console display. However, any background program can be dynamically swapped with the foreground program by the user. A hotkey, the default is ALT, is provided so that such a switch can be requested, or even a new window can be opened.

When DESQview is used with an 80386 based system, there is a companion product, QEMM, that will allow extended memory, memory above one megabyte, to be used as if it were expanded memory in the range 640K to one megabyte. Because of the limitations of DOS, program statements and data must reside in the first megabyte of memory when running. QEMM translates addresses above the one megabyte barrier to addresses below it by using special address registers available on the 80386 chip. This means that programs held in extended memory can be executed by setting a few registers instead of swapping the entire program with programs in storage below one megabyte. Thus there is essentially no overhead in task swapping. This product can be used on an 80286 system that has an expanded memory card.

Windows must be defined to DESQview before they can be accessed. An interactive program is included that assists in defining a window. Such information as full path name of the desired program, window size, whether graphics is required, and whether or not to continue to allow the program to run when it is swapped to the background, must be provided. Also it must be specified whether or not the program can be swapped out if the foreground task needs more memory. Caution is needed

```
R:COLOR.PIL
R:A PC-PILOT PROGRAM ON LANDSCAPE DESIGN

R:THIS SUBPROGRAM INITIALIZES THE TAPE UNIT
R:        Brings in Assembler routine to talk to Tape unit
DX:VIPC$(500)
FX:VIPCPI.BIN
FI:0,VIPC$
R:        If error condition found, message appears.
T(VIPC$(1)=CHR(255)):File VIPCPI.BIN not found!
R:       Change Address for EGA display
C:VIPC$(4)=CHR(002)
C:VIPC$(6)=CHR(003)
EC:

R:      This is the initialization screen.
TS:B0;F15
TX:
:
:
:
:          SYSTEM INITIALIZATION
:               IN PROGRESS
:
:
R:       Address of port for CGA is &H3C3
R:C:P=3C3
R:       Address of port for EGA is &H302
C:P=302
R:       Reset BCD board to start
C:VIPC$(9,80)="OUT   "!!P!!",0"!!CHR(13)
V:VIPC$

R:  THIS SECTION STARTS TAPE UNIT
R:   CHR(13) is carriage return and must be appended to commands
R:    to Tape unit.
C:VIPC$(9,80)="ECOF;TAPE;INIT"!!CHR(13)
V:ViPC$
W:15
V:VIPC$
W: 60
```

Figure 2. Pilot Control of VCR

here. Programs that control devices often do so through the use of an interrupt handler. Such a program should never be swapped out of memory since there is an interrupt vector pointing to it.

After all windows necessary for a particular learning experience have been defined to DESQview, the LEARN facility can be invoked to produce a startup script file that provides the student with a simple means of access. This script file will open all windows, and select the designated window as foreground. At any time, the student can access the other windows using the DESQview hotkey.

An Example

Consider an educational example on developing a research report that utilizes an encyclopedia contained on CD ROM, a word processor, a telecommunications program to provide access to a remote data base, and a PC Pilot program to direct student activity. The script file produced by the LEARN facility of DESQview would open all four windows and select the Pilot program as the foreground window. This program would present instructions to the student concerning the use of the system and give directions on the assignment. The student could look up information in the encyclopedia by using the DESQview hotkey to select that window as the foreground task. The remote data base could be accessed in the same way by selecting the communications window. Using the scratch pad facility of DESQview, the student could copy information from the encyclopedia or communications window into the word processor window to be included in a document or used as research notes.

A ballpark figure for the storage requirements of this configuration using commercially available full function software, is approximately 1.5 megabytes of memory. Thus a two megabyte 80386 based microcomputer should be sufficient to provide a system in which no paging is required.

Networking Multimedia Systems

The expense of a multimedia system makes it highly desirable to consider attaching the system to a network and making it available to all nodes of the network. There are several problems with this approach. The most obvious is that the LaserDisc and video players require a video monitor for their output. There is a reasonably simple solution to this problem. Since the output of these two devices is standard video, it can be made available at all nodes or be routed to a single very large video display if the application is in a laboratory. This suggests duplication of video monitors, but such an expense would be a very small part of the cost of a complete multimedia system.

A not so obvious problem is due to the very nature of this type of system. The software consists of various programs that perform completely separate functions, and were never designed to operate in conjunction with each other. Some of the programs used in a multimedia system control hardware that must be attached to the microcomputer on which they run. However, a standard method of distributing a program across the network is to load the program into the remote node from the network server across the network link, and then run the program on the remote node. This can not work for multimedia applications.

A hopeful way of dealing with this problem is to run the multimedia application on the network server and use a terminate-and-stay- resident (TSR) program [Stevens,Young] on the server to transport the various screen displays across the network and possibly accept keyboard from a remote node. This approach has the apparent advantage of not conflicting with current copyright law since the software involved in the application would always run on the server for which it was licensed. However, as yet only text transmission is available in commercial products. If an application requires graphics no help is currently available.

Bibliography

Quarterdeck Office Systems, DESQview. Santa Monica, CA.

Quarterdeck Office Systems, Quarterdeck Expanded Memory Manager-386. Santa Monica, CA.

Rhodes, D. M. (1987). Open learning in organizations through computer aided interactive video. In V. Hodgson, et al., Eds.., Beyond Distance Teaching: Toward Open Learning. London: SRHE/Open University Press, pp 86-93.

Rhodes, D. M. (1987). Multimediated CAIV: The Botany Program. Interactive Video Showcase, Association for the Development of Computer- Based Instructional Systems, Oakland, CA, November.

Rhodes, D. M. & Azbell, J. (1987). Authoring systems for interactive video instruction. In EURIT 86: Developments in Educational Software and Courseware (Proceedings of the First European Conference on Education and Information Technology). Oxford: Pergamon Press, pp 391-395.

Stevens, A. (1987). Turbo C: Memory-Resident Utilities, Screen I/O and Programming Techniques. Portland: MIS.

Washington Computer Services, PC/PILOT TUTORIAL MANUAL. Bellingham, WA.

Young, M. (1988, March/April). Converting a Turbo C program to a TSR. Programmer's Journal, 6(2), 50, 51, 53, 54, 56-62.

Special Presentation
Computers for Young Children

Harriet Paige
Irvine, California

Abstract

Outline of Presentation for Panel Discussion

Introduction: Speakers first exposure to computers and current use of computers in pre-school setting.

1. The computer as *another* educational tool. Not as a subject to be learned but as a tool to learn other subjects. Computers help children learn:
   - A. Concepts: example-opposites, colors
   - B. Specifics: example-numbers and letters
   - C. Social Skills: example-sharing, interaction with each other
   - D. Learning Skills: example-one thing at a time (Tortise and the Hare)

2. Computers meet needs of both learning types.
   - A. Visual learners have the screen
   - B. Rote learners can repeat the material at their own pace

   Both types can reinforce learning by using multitude of programs on the same subject

3. Helps the teacher organize the day, year, reference materials, and communication of staff, parents, etc..

Closing: The child who made it work.

Special Presentation
The V. I. C. Awards: Video in the Classroom Award Winning Applications

Hall Davidson
President, Video Using Educators
Los Angeles, California

Abstract

Abstract not available at press time.

Projects
Research—Computer Findings

Educational Computing: A Success Story or an Expensive Preoccupation

Andrew Tellep
Pennsylvania State University

Abstract

In light of the ever increasing need to use funds and resources wisely, it is particularly important that decision makers and practitioners have a clear understanding of the potential values of computers in education in order to gain such understanding, one must become informed of the successes and shortcomings of some attempts to improve education through computing and computing systems. This project was an *attempt* to bring together the results of many reports and empirical studies dealing with improvements, or lack of thereof, in education due to educational computing. The project was not confined to any particular level of education and includes results from work done at all levels except graduate level. After reviewing the report prepared at the end of this project, one should have a better view of the types of efforts that have improved education and what has simply been a preoccupation with hardware, software, labs, networks, etc. If our aim is to improve education and not to simply introduce students to computers, are we actually doing things in educational computing that can be show to be educationally sound and that cause positive effects in content areas i.e.—communication skills, problem solving, mathematics, science, social sciences? No matter how this question is answered, the answer is of vital importance to the professionals who will decide the future of educational computing.

University of Northern Iowa Computerized Middle School Curriculum Center

Philip Nelson
University of Northern Iowa

Abstract

Abstract not available at press time.

The Evaluation of Interactive Video Instruction

Sandra Levin
University of Illinois

Abstract

In the past decade, many have expounded the benefits of interactive Video (IV) in education, yet the little empirical research that has been done has come up with mixed results concerning the benefits of IV. This project presentation will briefly demonstrate the use of interactive video to teach earthquake education and will summarize the results of research conducted this past year on the use of IV with five different treatment groups. The results will focus on the effectiveness of IV and its impact on immediate and delayed achievement, motivation and attitudes, and classroom vs. lab setting.

Special Presentation

Logo as Intellectual Laboratory and Vehicle for Self Expression

Gary Stager
Wayne, New Jersey

Paper

Programming Variables: More Than Just a Name

Tronie Gunn
University of Houston, Texas

Abstract

Students do not automatically acquire a mature understanding of sophisticated computing concepts, such as that of variables, simply by experience in writing and editing simple programs. If students have only acquired immature concepts during instruction, their mental models of the underlying system may influence their concept development after instruction ceases. Specific mental models that are useful (such as the "container" model and the "slate" model) in developing the concept of computing variables, as well as specific mental models that are counterproductive (such as the "names" model, the "smart computer" model, and the "black box" model) are characterized.

The concept of variable is both inconstant and elusive. It is hard to describe (Schoenfeld & Arcavi, 1988) and hard to learn (Rosnick, 1981). A survey of definitions of variable reveals that it means different things in different contexts. Understanding the mathematical concept of variable provides the basis for the transition from arithmetic to algebra and is necessary for the meaningful use of all advanced mathematics. In the same way, understanding the computing concept of variable is the key to the production of computer programs that have the desirable property of implementing general algorithms that can be applied to a variety of data.

From the programming point of view a variable is defined as an address or memory location for which a name or label (generally a letter or string of letters and/or other characters, depending on the language) has been designated. The portion of a program in which a given name or label is accessible is called the scope of that variable, although it is more properly the scope of the variable name. Variable names are often accessible throughout an entire program; these variables have "global" scope (for example, all variables in many versions of BASIC). In other programming languages (such as Logo), some variable names may be accessible only within specific procedures and not within other procedures that comprise a single program; these variables have "local" scope.

At any particular time, a variable name will reference a memory location that contains a number (or numerically encoded character string), called the value of the variable, which may change from time to time. Because every memory location contains some bit pattern, and any bit pattern present in a memory location may be interpreted to represent some value, a variable cannot ever be truly "valueless." Some languages distinguish between variables that have explicitly assigned values and those that do not. They identify variables with no explicitly assigned value as "valueless." In many situations, a value for the variable must be designated, implicitly or explicitly, when a variable is created. The value of a variable may change millions of times during the execution of a program. However, when the value of a variable changes, its location, naming, and functional relationships with other elements of the program are invariant.

In many uses of programming variables, the mathematical concept of variables and equality would be sufficient to determine the result of a programming instruction. For example, for assignments that only depend on constant values and/or other variables, a unique value can be associated with the name of the assigned variable. However, the understanding of the incrementation/decrementation assignment makes it necessary to consider a temporal aspect: the value of the variable after the assignment is replaced by the value of the variable before the assignment, plus or minus some value. The student must designate by the same name both the preceding value and the present value (which depends on the former), and has to process the assignment as an asymmetric relation. It seems likely, therefore, that understanding this type of assignment will be extremely difficult for novices.

Subjects of all ages have demonstrated misconceptions related to the concept of programming variables. Some of the most frequent misconceptions have been an assumed relationship between the variable's name and its value or function in the program and misunderstandings related to the passing of parameters and the scope of variables between and within procedures.

All teachers have a vital role in guiding the development of the concept of programming variables and other important computing concepts in their students. But what do teachers know about programming variables after the brief introduction allowed in a typical computer literacy overview course? What mental models might they use to guide the future development of their concept of programming variables after they have completed formal instruction?

The Study

An exploratory study, conducted in the spring of 1990, explored the mental models acquired by preservice and inservice teachers in the context of computer literacy Logo instruction. As part of the study, the subjects' concepts of computing variables were probed in interviews. This report focuses on the results of those interviews.

Methodology

Subjects were 13 student volunteers enrolled in sections of the undergraduate and graduate computer literacy for teachers courses offered at the University of Houston during the Spring Semester, 1990. They participated in the routine course introduction to the Logo computer language with their class sections, completing the same assignments and activities as other students enrolled in the courses. To insure comparability between the Logo experiences of the graduate and undergraduate students, the instructor for the graduate course presented the lecture on variables for both courses. The student volunteers were asked to complete a more detailed experience and demographic questionnaire than that routinely used in the courses. Based on that information, subjects were selected for diversity along several dimensions: (a) classification, (b) gender, (c) teaching experience, (d) programming experience, (e) major field, (f) self-reported computer anxiety, and (g) age.

Subjects were then asked to complete three additional tasks: (a) a booklet in which they were asked to write down in ordinary English "what happens inside the computer when ..." a variety of Logo commands are entered (similar to the booklet developed for BASIC by Bayman & Mayer, 1983), (b) a videotaped interview in which they were questioned about their understanding of "variable" as a computing concept, and (c) a series of brief Logo commands and procedures for which they were asked to predict and explain the results of the commands. This paper will briefly report the results of the first and third tasks, while focusing on the second. Questions asked in the "variables" interviews included the following (with additional clarifying probes, as needed):

1. What is a variable?
2. Does every variable have a name?
3. Give an example of a name which a Logo variable could have.
4. How many names can a given variable have? Could a variable have more than one name at any particular time? Why?
5. When and how does a variable get a name?
6. Can the name of a variable be changed? If so, how?
7. Does every variable have a value? Explain.
8. Give an example of a value which a Logo variable could have. Give an example of a value which a Logo variable could not have.
9. How many values can a given variable have at one time? Explain.
10. When and how does a variable get a value?
11. Can the value of a variable be changed? If so, how?

Results: The Logo Booklet

Three categories of language statements that were studied included direct assignments (using MAKE), output (using PRINT), and repetition (using REPEAT). The subjects tended to verbalize conceptions of the statements that either failed to include the main idea or that included outright misconceptions for all three categories of statements. For example, the location in which a value obtained by incrementation would be stored (replacing the variable's previous value) was not clear and subjects commonly indicated that all assignment statements also produced some form of visible output. Subjects often failed to mention the repetition when trying to describe the repetition of an assignment than when trying to describe the repetition of printing. Although all subjects had demonstrated "hands-on" competence in using Logo, this experience was not sufficient for the mature understanding of programming.

Results: The Logo Prediction/Explanation Examples

The errors and misconceptions observed in the prediction/explanation portion of the study were of several types: incomplete/incorrect knowledge, apparent inconsistencies of the Logo environment, and attribution of human-like reasoning capabilities to the computer. Many errors were similar to those observed in the Logo booklet transaction lists. These include a variety of difficulties with assignments (especially incrementation/ decrementation) regarding related output and their use in repetition.

Of the errors not obviously related to the transaction lists, perhaps the most common were those that attribute to the computer human-like will or understanding. Subjects who suggested that the actions of the computer were not within programming control or that the computer could understand the names of procedures and/or variables and act accordingly evidence counterproductive mental models. High anxiety subjects, subjects whose major fields were not mathematics or science, and subjects with no programming background were the most likely to make these errors.

Results: Transactions and Predictions

There was a statistically significant Pearson correlation of $r = .68$ ($p < .05$) between the number of incorrect/incomplete conceptions in the Logo booklet transaction lists (possible misconceptions) and the number of examples presented before no later examples were predicted incorrectly. This is equivalent to the correlation between the number of correct conceptions and the latest number of consecutive prediction examples that were at least partially correct.

Results: The "Variable" Interview

Subjects generally exhibited relatively incomplete concepts of variables. Several subjects demonstrated some mixing of the ideas of variables and procedures in Logo. Results of the responses to questions in the interview are summarized in Table 1.

Table 1.
Components of a Mature Concept of Variable Observed in "Variables" Interview

Variables have:	Optional	Required
* a name	100.0%	83.3%
* a value	100.0%	58.3%
* a location	33.3%	33.3%
* a scope	0.0%	0.0%
Variable characteristics:	Name	Value
* can be changed	83.3%	100.0%
* only 1 at a time	75.0%	91.7%
* valid example	91.7%	91.7%
Assignment by:		
* MAKE command		16.7%
* TO command		16.7%
* other or unknown command		58.3%

The subjects generally presented naive but reasonable descriptions of variables as placeholders or substitutes (similar to the mathematical concept of variable). They identified some defining characteristics of variables but omitted others. For example, all agreed that variables could have names and values and most indicated that variables were required to have both. Only a few suggested that variables could have a location, but those subjects indicated that a location was a requirement. No subjects brought up the characteristic of scope (not particularly surprising for novices!). It is the lack of inclusion of location as a critical, required component of a computing variable that may have been responsible for many of the observed misconceptions.

The subjects generally agreed that variables could have only a single name and a single value at any time. They could usually offer valid examples of both names and values. They agreed that the values of variables could be changed, although they were usually unable to specify the command for doing so. Most also said that the name of a variable could be changed, which is not possible within Logo. Given the commonly identified mechanism for accomplishing this change (editing a command or procedure), they may have meant that the name of a variable was arbitrary and subject to the programmer's choice. The subjects were generally uncertain about what creates a "new" variable and what happens to "old" variables when they are no longer in use.

Mental Models

"Mental models are the mechanisms whereby humans are able to generate descriptions of system purpose and form, explanations of system functioning and observed system states, and predictions of future system states." (Rouse & Morris, 1985, p. 7) Although the variables interview was intended to probe the subjects' concepts of programming variables at that time, the subjects' mental models of variables (and of computing in general) would likely influence the future development of those concepts after the subjects are removed from the guidance of a course instructor.

Subjects seldom articulated a clear picture of the mental models they may have been using to make their predictions and explanations. However, based on the patterns of errors and misconceptions, it is possible to infer broad characterizations of the mental models that may have been used. No claim is made that these models are the only ones that could have been in use.

Several mental models that may have been employed did not relate directly to variables. Instead, they related to the subjects' overall view of computers and problem-solving. These include (a) the "smart computer" model, in which the computer or the currently executing program is expected to possess intelligence, understanding, and will; (b) the "black box" model, in which the functioning of the computer is considered mysterious and not easily comprehensible; and (c) the "use everything" model, in which problem solving is viewed as simply combining as many as possible of the apparent facts into the "right" form. These models are highly counterproductive, directing the attention to features that are only coincidentally related to the actual results. Some subjects may have used combinations of these models, no models at all, and/or other, more variable-specific models.

Some of the possible mental models used related directly to the subjects' concept of computing variables. These models include the "container" model, the "slate" model, the "names" model, and the "procedures" model. Each of these is explained in more detail below.

The "Container" model. In this model, subjects view variables as named containers for values. This type of model is among the most commonly suggested models for variables in textbooks and typical programming instruction (e.g., Fox & Waite, 1981; Luehrmann & Peckham, 1983). Programming instruction typically stresses that these containers (variables) can only hold a single value at a time. However, students may imagine that the containers (variables) can hold many values at once, just as the real-world containers with which they are familiar can hold many things at one time (DuBoulay, 1989). Although imagining potentially empty containers in a situation such as Logo programming that does not allow empty variables to exist is unlikely to cause difficulties in making predictions or explanations, it is another possible shortcoming of this model.

An important potential shortcoming of this model becomes obvious when the possibility of both local and global variables is considered. To account for the possibility of both local and global variables, it is necessary to allow for multiple sets of containers to exist. The containers within each set would be required to have unique names, but containers in two different sets could have identical names, because the set to which the container belonged would be a distinguishing identifier. There also would be one set designated as the "global" set, which would become, in effect, joint property of all the "local" sets of containers and would be referenced if no "local" container had an identical name.

One subject who consistently referred to an analogy of variables with boxes obviously used a containers model. This model was clearly useful for her. Other subjects may have also used this model.

The "Slate" model. This model was not directly observed as the only possible explanation of the predictions of any subjects, but may have been the underlying model of many successful subjects. In this model, computer memory is viewed as containing an erasable slate on which the names and current values of variables can be written. A memory reference copies the contents from the slate, while an assignment writes a new value, after erasing the previous one, if any. Subprocedure calls can be correctly viewed as not simply passing control to a new set of commands (as in BASIC), but also making a set of variable assignments based on the current values of specified variables (or constants). Subjects with this model would have no difficulty with multiple assignments - the slate can only have one value (the latest assigned) for a given variable.

The "Names" model. Subjects with this model seem to view variables as location-less abstractions, similar to mathematical variables. These variables can have many values, but only one at a time (for a given exercise). However, the value is linked to the name and the name is globally known (at least for the duration of that exercise). This model seems to be basically parallel and static. A temporal dimension is definitely missing.

This model causes severe problems in encounters with incrementation/decrementation assignments. It is not reasonable to consider that C can in any way be or become C+1. Though Logo uses a command (MAKE) that suggests the temporal nature of the assignment, instead of the potentially more confusing equal sign used in some languages, subjects using this model were still probably confused.

Another error that may have come from this model involves difficulties with repeated assignments. Making two incompatible assignments (such as making TIMES both 15 and 25) is incredibly confusing when processed through the names model.

This model seems unlikely to evolve into a more adequate model without knowledgeable intervention. Subjects have a wide range of examples in mathematics upon which to base this model. The limited exposure to programming in an overview course is unlikely to offer sufficient examples to discard this model, which worked successfully in other contexts. The instructor must clearly present and explain examples for which this model is insufficient and misleading.

The "Procedures" model. Subjects with this model may have overgeneralized their models of procedures in Logo to use them as models for variables. This could result in the assumption that incrementation/decrementation assignments would create new copies of the variables named in a manner similar to recursion of procedures. As there is no command-based equivalent of in-place replacement of a procedure in Logo, generating an explanation of this type of command could be difficult from this model. Once a procedure is defined, the definition is stored ("remembered") and is later invoked by naming it. This is similar to the expectation of several subjects that assignment equations are remembered and can reproduce the specified action later by name. Object Logo does not allow the user to define a procedure with the same name as one already defined in the workspace; an error message results. Some subjects suggested that repeated assignments would also result in an error message or that an assignment to a variable called W might cause a problem because of the already defined procedure named W.

Conclusions

Subjects did not exhibit mature concepts of programming variables, although they typically did not evidence strong inaccuracies in their naive concepts. Given appropriate time and opportunities to tune their developing concepts, the subjects could be expected to develop mature concepts of programming variables. If most will not have additional formal instruction regarding programming variables, their mental models (if they have formed one) could be expected to guide any further development of the concepts of programming variables.

Subjects with "smart computer" and "black box" models could be expected to have limited future development of their concepts of variables without further instruction. Their mental models of computing lead them to feel that they cannot control or understand what happens, so they make little effort to do so.

Both the "container" and "slate" models (and to a smaller degree, the "procedures" model) usually support good predictions about the functioning of Logo variables. Instructors could be encouraged to teach these models of variables, with their limitations. Experience with the Logo environment in particular and with computing in general might sufficiently identify and clarify limitations of each model (without further formal instruction) so a mature concept of programming variable can result (perhaps without the component of scope, unless specific knowledgeable intervention occurs).

Subjects utilizing the "names" model seem to be doomed to experience confusion and frustration. Subjects can readily construct this mental model based on years of exposure to it through mathematics. Unfortunately, the fundamental natures of mathematical variables and computing variables are different! Mathematical variables are abstractions - they can only be identified by their names. But computing variables are more than just a name! Computing variables have a location - the name is really only a convenience for the programmer. Computing variables with the same names can assume different functions within the context of the same program and computing variables with different names can communicate values from one part of a program to another. Care should be taken to eliminate this model from the repertoire of all computing students, but especially those who plan to become teachers and who might pass this model on to other students.

Bibliography

Bayman, P., & Mayer, R. E. (1983). A diagnosis of beginning programmers' misconceptions of BASIC programming statements. Communications of the ACM, 26, 677-679.

DuBoulay, B. (1989). Difficulties of learning to program. In E. Soloway & J. C. Spohrer (Eds)., Studying the novice programmer (pp. 283-299). Hillsdale, NJ: Lawrence Erlbaum.

Fox, D., & Waite, M. (1981) Pascal primer. Indianapolis: The Waite Group, Howard W. Sims.

Luehrmann, A., & Peckham, H. (1983). Computer literacy: A hands on approach. Mission Hills, CA: Glencoe/McGraw-Hill.

Rosnick, P. (1981). Some misconceptions concerning the concept of variable. The Mathematics Teacher, 74, 418-420.

Rouse, W. B., & Morris, N. M. (1985, May). On looking into the black box: Prospects and limits in the search for mental models (Report No. ONR-TR-85-2). Arlington, VA: Office of Naval Research. (ERIC Document Reproduction Service No. ED 268 131)

Schoenfeld, A. H., & Arcavi, A. (1988). On the meaning of variable. The Mathematics Teacher, 81, 420-42

Paper

Teaching the Idealizations: Altering Science Alternative Conceptions with Microcomputer Simulations

Herman G. Weller
Texas Tech University

Norman R. Dodl
Virginia Tech

Abstract

A large part of the science student's prior knowledge, which has been derived from the individual's previous experiences, is not compatible with what is taught in science classes. The naive models of the natural world that the child brings to the classroom are often quite resistant to traditional teaching. If both the teacher and the student cannot detect and change these naive models, or *alternative conceptions*, the student's ability to continue learning science may be seriously impeded.

This is a report on the development and investigation of two microcomputer-displayed simulations of *idealized scientific phenomena* that were designed to facilitate the 8th-grade science student's alteration of three Aristotelian alternative conceptions. The instructional strategies incorporated into the computer simulations were consistent with a theory of science conceptual change that: a) facilitated the student's recognition and discovery of a phenomenon that was contradictory to his or her conceptual framework and that was typical of the corresponding scientific concept, and b) allowed the student to manipulate the objects and relationships of the phenomenon, experiencing the consequences of that action, so that the student would adjust his or her conceptual categories until the phenomenon became anticipated.

Eighth-grade students who were selected for possession of alternative conceptions were helped by the computer simulations to alter their naive conceptions to a significant degree. This was shown by a post-test administered immediately after the students had worked through the simulations, and by a retention test administered to a sample of the students 1.5 months later.

Introduction

For many children, science is a difficult subject to learn. A large part of the child's prior knowledge, which has been derived from the world of everyday experience, is not compatible with what is taught in science classes. The naive models of the natural world that the child brings to the classroom are often quite resistant to traditional science teaching. If both the teacher and the student cannot detect and change these naive models, or *alternative conceptions*, the student's ability to continue learning science may be seriously impeded.

One solution to this instructional problem would be to construct a microcomputer-based system that a teacher could use a) to diagnose a student's possession of alternative conceptions, and b) to help the student develop a conception more in line with the current scientific viewpoint to replace each alternative conception. This is a report on the development and investigation of the remediation component of such a system: two microcomputer-displayed simulations of *idealized scientific phenomena*, which place the student in environments where a) he or she can *play the game of science* in contrived situations where his or her naive conception is at conflict with the displayed phenomenon, and b) he or she can experiment with examples of the more "scientific" concept until the conflicting phenomenon becomes the anticipated phenomenon.

Alternative Conceptions

For over twenty years, science education research has been accumulating evidence that a significant number of science students of all ages, weak as well as strong learners, possess alternative conceptions (Hasweh, 1988). These are the alternative views of the world that have been derived from the individual's previous experiences. They often persist in spite of several years of traditional instruction in concepts that are more in line with the scientist's viewpoint and that contradict the naive conceptions. In many cases, these naive models of the natural world have been found to occur regularly across different students. In the area of force and motion, many present-day alternative conceptions have been found to be similar to early explanations (Hasweh, 1988).

The students' possession of alternative conceptions of force and motion should not come as a surprise to the science educator, for the world of force and motion in everyday experience is not identical to the science models of the classroom that have been abstracted from the natural world. The world of everyday experience is indeed largely an Aristotelian world (Garrison & Bentley, 1990). For example, the child's world outside of school is not airless, frictionless, or without gravity. Heavier objects do fall more quickly than lighter ones (say, a stone and feather falling through the air). Sliding objects do require forces to keep them in motion (say, a sled on lake ice) and the retarding friction is not often recognized as a force.

It is often in the physical science course in the eighth or ninth grade that many students first encounter extensive formal instruction that contradicts several of their naive conceptions of force and motion. In spite of the instruction there and in the grades following, the same alternative conceptions are also held by secondary and university students. For example, Champagne, Klopfer, and Anderson (1980) found that most of the college students they surveyed believed that objects fall at a constant speed, a speed that the students thought (as had Aristotle) to be proportional to the weights of the objects. Because the eighth-grade confrontations between students' naive dynamics beliefs and instruction on the scientific viewpoint have significant importance for the children's future science learning, the present investigation concentrated on fostering conceptual change of eighth-graders who were found to possess Aristotelian alternative conceptions of force and motion. These were eighth-graders who were found to hold one or more of the following three alternative conceptions several weeks after their physical science classes had covered force and motion.

1) *Quickness of fall depends on mass* conception
   With no air resistance, the downward movement of a more massive object is faster than that of a less massive object.
2) *Unbalanced force* conception
   If an object is moving and no unbalanced force is acting on it, the object will come to rest.
3) *Direction of vertical movement depends on type of material* conception
   a) Light objects rise because they are made of material that naturally tends to move upward;
   b) heavy objects fall because they are made of material that naturally tends to move downward.

Physical Science: An Alien World of Abstractions?

Most students, and too many science teachers, do not realize that physical scientists have commonly dealt with the staggering complexity of the natural world by constructing idealized models of various real-world phenomena. Hence, one typically finds such statements in eighth-grade science as 1) "When objects of different mass are dropped simultaneously from the same point *in a vacuum*, they strike the ground at the same time" and 2) "Objects will continue moving indefinitely unless they are affected by some *outside influence*." However, in his or her experience, the student has not dropped objects *in a vacuum* and has not moved objects where they were unaffected by *outside influences*.

Unfortunately, many science teachers (and perhaps some physical scientists) believe that the abstract models of science determine the natural world, rather than follow from it. Certainly, for many students, the *network of idealizations* that is physical science, where one explains the natural world by referring it to mathematical formulae, is indeed an alien world with an alien language. In the authors' view, a theory for changing a student's naive conceptions in science should take account of the fact that the corresponding scientific conceptions are often abstractions from the world of practical experience, and should allow for student practice with examples of the appropriate idealized scientific situation.

A Model for Changing the Science Student's Conceptions

Hasweh (1988) has contended that a conceptual change study should rest upon the foundation of a specified, appropriate theory of conceptual change. Instructional strategies can then be proposed and implemented in order to test the theory.

The first author (Weller, 1990) has constructed a theory for instructionally eliciting a student's conceptual change in science. The crux of the theory is that the science student who has an alternative conception can be helped to discover a phenomenon that is contradictory to his or her conceptual framework and that is typical of the relevant scientific concept. The teacher can accomplish this by allowing the student to examine the phenomenon repeatedly, in effect *playing* with examples of the scientific concept by acting upon the objects and relationships of the phenomenon and then experiencing the consequences of that action. The result is that the student gradually adjusts his or her conceptual categories until the phenomenon becomes anticipated.

For the construction of this conceptual change theory, Weller (1990) has drawn heavily upon 1) the theoretical views of Thomas Kuhn, both on the role of anomaly in the discovery of a new and unsuspected phenomenon (Kuhn, 1970) and on the function of *thought experiments* in science (Kuhn, 1977), and 2) the thoughts of John Dewey (1916) on the relation of experience to thinking. The conceptual change theory has drawn also upon a) the model for the facilitation of conceptual change in science that was proposed by Posner, Strike, Hewson and Gertzog (1982), and b) the criticism of the latter model by Garrison and Bentley (1990) for its assumption that a student's learning is a *rational* activity, and for ignoring *prerational* considerations in learning.

Weller (1990) has described an individual learner's process of conceptual change in terms of the three stages that Kuhn (1970) has portrayed in a scientific community's discovery of a new and unsuspected phenomenon: the practice of normal science, awareness of anomaly, and conceptual adjustment. To help the student become aware of anomaly, Weller has employed Dewey's (1916) notion of an individual's learning experience as an active-passive affair, in which "Experience as trying involves change, but change is meaningless transition unless it is consciously connected with the return wave of consequences which flow from it." The student learns when he or she acts upon something and then undergoes the consequences of that action.

Posner et al. (1982) have proposed that for radical conceptual change to occur the learner must be dissatisfied with his or her existing conception and the new conception must be intelligible, initially plausible, and apparently fruitful for future use. Garrison and Bentley (1990) have maintained that Posner's description is incomplete because learning for the beginning science student takes place before the individual has formed complete "scientific" concepts and categories. To this student the idealized system of science represents another world, with standards of what constitutes valid reasoning and sound evidence that are entirely different from the world of everyday practice.

Weller (1990) has listed the following as necessary characteristics of the instructional situation that presents the contradictory phenomenon to the student:

1. It presents the scientific idealization of the contradictory phenomenon.
2. It exhibits cues that the student usually employs when making judgments in this area.
3. It confronts the student with a contradiction.
4. It provides the student with enough control over the situation that he or she can restart the contradictory phenomenon several times.
5. It omits distracting variables that are not important for understanding the anomalous phenomenon.
6. The conflict deduced by the student from the contradictory phenomenon must be one that the student could confront in experiencing nature.
7. The deduced conflict must be one that, however vaguely perceived, has confronted the student before.

The Microcomputer Simulations

Simulation is the "process of interacting with a model that represents reality" (Lunetta & Hofstein, 1981). According to Gagne (1962), a simulation a) represents a real situation in which operations are carried out, b) provides the user with certain controls over the problem or situation, c) omits certain distracting variables that are irrelevant or unimportant for the particular instructional goals.

There have not been many rigorous studies to date of the validity and effectiveness of computer simulations in physical science instruction. Twelker (1968) has suggested that the degree to which a simulation corresponds with the real-world situation it is designed to simulate may be crucial to learning. He felt that if there is a credibility gap between instruction and the operational world, then the learner may be at a disadvantage when it comes to performing in, and/or understanding, the real world. This view does not take into account the fact that the world of science is actually an abstraction from the world of practical experience.

In an alternate view, the authors maintain that for a simulation designed to induce conceptual change, strict adherence to the conditions of a real-world situation is much less important than it is to present the scientific idealization of the real-world situation wherein a phenomenon obviously contradicts the student's naive conception(s). Such an idealized situation strongly resembles the *thought experiment* of the scientist. A *thought experiment* is one that is imagined by the scientist but not actually carried out. It may involve a situation that has not been examined in a laboratory, or even a situation that could not possibly occur in nature (Kuhn, 1977). A well-known example is Galileo's *thought experiment* where he investigated the dependence of an object's acceleration on its weight by imagining two small equal weights falling separately side by side or chained together. Weller (1990) holds that the conditions necessary for facilitating a student's conceptual change with a simulation are the same as the conditions proposed by Kuhn (1977) as being necessary for the effectiveness of *thought experiments* in eliciting conceptual change.

Arguing that from thought experiments most people learn about both their own conceptions and the world, Kuhn (1977) held that a thought experiment must be capable of confronting the thinker with a paradox of such impact concerning the person's incorrect conceptions, "transforming felt anomaly to concrete contradiction," that the person is both informed that he or she is wrong and provided with clues necessary for *reforming* the conceptions. Concerning the conditions necessary for this impact, he concluded first that although the situation imagined in the thought experiment need not even be potentially realizable in nature, it must exhibit the cues which the thinker usually employs when making judgments in this area. Secondly, the conflict deduced from the imagined situation must not only be one that nature itself could present, but must also be one that, however, vaguely perceived, has confronted the thinker before.

The two computer simulations in the present study were designed to facilitate the student's conceptual adjustment in the following ways. Each simulation focused the student's attention on a phenomenon that was a scientific ideal case and that was in contradiction to one or more of his or her diagnosed alternative conceptions. The student reran the first simulation ten times and the second simulation five times, in order to have time a) to become aware that the highlighted phenomenon was contradictory to his or her conceptual framework and b) to adjust his or her conceptual categories until he or she came to expect the phenomenon each time. Each simulation exhibited cues by which the student usually judges the motion of objects. Because the student's conceptual change might be accompanied by student resistance, each simulation was designed to abate many feelings of resistance to discovery by employing appropriate principles of human-computer interactivity (Weller, 1988) to a) make the student feel in control of the simulation, and b) provide him or her enjoyment and satisfaction in running the simulation.

Objects Falling/Rising simulation

For the *Objects Falling/Rising* simulation (Figure # 1), the student chose the materials from which two objects were composed and the height on a tower from which those objects fell. The student also chose the mass of each object, unless the

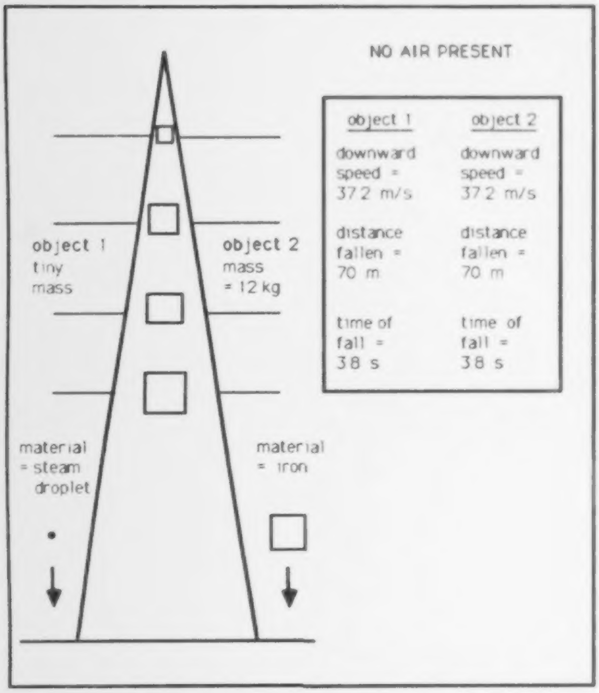

NO AIR PRESENT
object 1
downward speed = 37.2 m/s
distance fallen = 70 m
time of fall = 3.8 s
object 2
downward speed = 37.2 m/s
distance fallen = 70 m
time of fall = 3.8 s
object 1
tiny mass
object 2
mass = 12 kg
material = steam droplet
material = iron

Figure 1. The *Objects Falling/Rising* simulation. Two objects fall or rise on opposite side of a tower. Air is alternatively present or not present in successive runs of the simulation.

object was a steam droplet or a dust particle (whose masses were merely stated as "tiny"). The simulation runs were alternately in an environment of *air resistance* or *no air resistance*. In the cases when the objects fell from the tower to the ground, the right half of the screen showed at different times a) no supplemental information, b) two downward arrows representing analogously the current speed of fall of the objects, c) the increasing times of fall, speeds, and distances fallen of the objects. In the cases when the objects did not fall (i.e., in the case of the steam droplet or the dust particle in an environment of air resistance), the supplemental information was not provided.

When the objects fell with no air resistance, they moved at opposite sides of a tower and struck the ground at the same time. Both the parallel fall of the objects past tower stories and their simultaneous impact with the ground presented cues that a person customarily employs when making judgments about falling objects.

The *Objects Falling/Rising* simulation was designed to provide, in the case of no air resistance, a situation in direct conflict with the beliefs of the student who had a *quickness of fall depends on mass* alternative conception (i.e., who believed that, with no air resistance, the object with more mass will strike the ground first). The conflict was one that can occur in nature when objects much denser than air are dropped within a few meters of the earth, and one that the student had quite likely encountered before (even if only vaguely perceived). The simulated fall of the two objects, as well as the supplemental information, provided clues to allow the student to work toward altering his or her conception.

The *Objects Falling/Rising* simulation was also designed to provide, via the contrast of the air-resistance and no-air-resistance environments, situations in direct conflict with the beliefs of the student who had a *direction of vertical movement depends on type of material* conception (i.e., who believed that light objects rise because they are made of material that naturally tends to move upward, and that heavy objects fall because they are made of material that naturally tends to move downward).

Cube Moving simulation

For the *Cube Moving* simulation (Figure # 2), the student chose the mass and initial horizontal velocity of a cube. For each simulation run, the computer system chose four new forces that were exerted on the cube throughout its motion (an upward force, downward force, force from the left, and force from the right). While the cube moved across the screen, the forces upon the cube were represented by arrows with lengths proportional

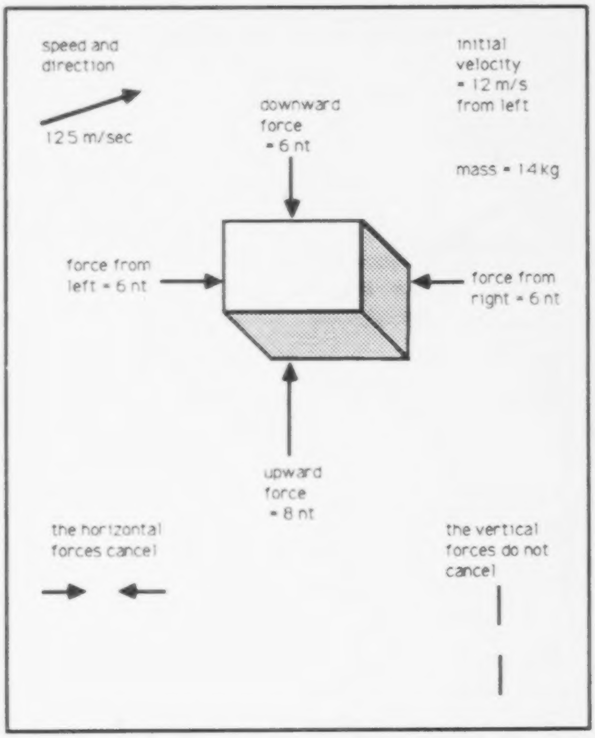

speed and direction
12.5 m/sec
initial velocity = 12 m/s from left
downward force = 6 nt
mass = 14 kg
force from left = 6 nt
force from right = 6 nt
upward force = 8 nt
the horizontal forces cancel
the vertical forces do not cancel

Figure 2. The *Cube Moving* simulation. As the cube moves, the incident forces and the current speed and direction are always shown. After the start of the simulation, the initial velocity, mass, and force-cancellation diagrams disappear.

to their magnitudes. During the motion, an arrow in the upper left portion of the screen represented the current velocity of the cube. The magnitude of the velocity was printed alongside the arrow.

Before the *Cube Moving* simulation started, two diagrams at the bottom of the screen informed the student whether the horizontal forces on the cube were balanced and/or whether the vertical forces were balanced. The motion of the cube relative to the screen borders, and to the cube's initial starting point, presented cues that a person typically uses when making judgments about moving objects.

Whenever the four forces were balanced in the *Cube Moving* simulation (i.e., when the resultant force was zero), the cube did not come to rest. Instead, it continued to move with its initial velocity. This situation was in direct conflict with the beliefs of the student who had an *unbalanced force* conception (i.e., who believed that a moving object on which the forces are balanced will come to rest). The conflict was one that could occur in nature when massive objects moved across low-friction surfaces (such as ice), or when objects moved at large distances from the earth (which the student might have observed before in films of astronauts on board a satellite). It was also a conflict with which the student had probably been confronted before. As with the objects in the *Objects Falling/Rising* simulation, the motion of the cube, as well as the supplemental information, provided clues to allow the student to work toward altering his or her conception.

Methodology

A series of interventions was implemented with a representative sample of students drawn from eighth-grade physical science classes at two schools, with 29 and 31 students, respectively. The first school was a suburban private school for grades K-12; the second was a high school that drew students from a large rural area containing several large towns. The 60 students from these two schools had studied force and motion in their science classes two and six months earlier, respectively.

When students were detected by a pre-test as possessing alternative conceptions, the system presented the two simulations, which were designed to facilitate the student's alteration of one or more of these naive conceptions. The control group students did not do the simulations. They filled out questionnaires concerning their attitudes and opinions about the pre-test, while the experimental group students worked through the two simulations. We determined evidence of immediate change by administering a post-test both to students who had worked the simulations and to students who had not. We assessed evidence of the students' retention of simulation-elicited changes 1.5 months later by readministering the post-test to a sample of those students who had worked the simulations.

Results and conclusions

The students who were selected for possession of alternative conceptions were helped by the computer simulations to alter their naive conceptions to a significant degree. A post-test supplied evidence of the students' short-term conceptual change. The students in the experimental group answered an average of 5.9 questions correctly (out of eight) on the post-test, with a standard deviation of 1.5 questions; the control group students answered an average of 4.8 questions correctly, with a standard deviation of 1.8. A *Students t test for independent samples* showed that the experimental mean was significantly superior to the control mean ($t=2.64$, $p=0.005$).

The retention test that was taken by a sample of seven students 1.5 months later supplied evidence of robustness of the conceptual change. Considering the specific question that each of six of the students had originally answered *nonscientifically* on the computer diagnostic test and *scientifically* on the computer post-test, the six students each answered the same question scientifically on the retention test. The seventh student, who had originally answered two more questions *scientifically* on the computer post-test than on the computer diagnostic test, answered one of the two questions *scientifically* on the retention test.

Educational Importance of Study

These computer simulations could serve as a prototype remediation system for similar microcomputer simulations that would deal with many of the well-characterized alternative conceptions known to occur in the learning of physical science in grade eight (or nine). Such a remediation method would not require the input of considerable additional time and effort by the science teacher and would not require the purchase of specialized laboratory apparatus designed for the remediation of each different alternative conception.

Although the physical science student has not grasped completely the standards of what constitutes valid reasoning and sound evidence in the idealized system of science, computer simulations can help him or her to learn the canons of scientific rationality by *playing the game of science*. The student does this by encountering examples of situations that scientists have already learned to view as being different from other situations (e.g., ideal gas laws, free fall in a vacuum, and motion along frictionless surfaces). In this way he or she can build up a supportive web of familiarity with exemplars and nonexemplars of scientific thinking which will allow him or her to surmount the barrier of abstraction, judgment, and emotional acceptance between the naive concepts and the science concepts.

Bibliography

Dewey, J. (1916). *Democracy and education*. New York: The Free Press.

Champagne, A. B., Klopfer, L. E., and Anderson, J. H. (1980). Factors influencing the learning of classical mechanics. *American Journal of Physics*, 48, 1074-1079.

Gagne, R. (1962). The use of simulators. In R. Glazer (Ed.), *Training research and education*. Pittsburgh: University of Pittsburgh Press.

Garrison, J. W., & Bentley, M. L. (1990). Science education, conceptual change and breaking with everyday experience. *Studies in Philosophy and Education*, 10(1), 20-35.

Hasweh, M. (1988). Descriptive studies of students' conceptions in science. *Journal of Research in Science Teaching*, 25(2), 121-134.

Kuhn, T. S. (1970). *The structure of scientific revolutions*. Chicago: University of Chicago.

Kuhn, T. S. (1977). A function for thought experiments. In T. S. Kuhn (Ed.), *The essential tension* (pp. 241-265). Chicago: University of Chicago.

Lunetta, V. N., & Hofstein, A. (1981). Simulations in science education. *Science Education*, 65(3), 243-252.

Posner, G. J., Strike, K. A., Hewson, P. W., & Gertzog, W. A. (1982). Accommodation of a scientific conception: Toward a theory of conceptual change. *Science Education*, 66(2), 211-227.

Twelker, P. A. (1968). Simulations: An overview. *ERIC document*: ED25459.

Weller, H. G. (1988). Interactivity in microcomputer-based instruction: Its essential components and how it can be enhanced. *Educational Technology*, 28(2), 23-27.

Weller, H. G. (1990). *Microcomputer-based diagnosis and remediation of simple Aristotelian conceptions of force and motion*. Unpublished doctoral dissertation, Virginia Polytechnic Institute and State University, Blacksburg, Virginia.

Special Presentation—ISTE

Responsible Computing: Computers, Ethics and Students

Abstract

Although microcomputers have been in classrooms for at least a decade and we have witnessed marvelous gains in the power of computers and the quality of software, relatively few gains have been made in our understanding of the ethical use of computers and information. Increased access to information resources in electronic form has created a host of new ethical problems and has changes the context in which a number of ethical issues arise.

The panel addresses a number of concerns and issues in computer ethics from an educational and law enforcement perspective. It also suggests educational strategies that may be employed to help develop responsible computing in students. The panelists all presently serve as members of the Advisory Council of a Joint National Institute of Justice—U.S. Department of Education Computer Ethics project designed to help develop responsible use of computers by students.

Panelists:

Chairperson:
Jonathan Budd
Director, Computer Ethics Project
Program Manager, Law Enforcement, Computer Crime
National Institute of Justice
Panel Chair

Steve Purdy
Agent, U.S. Secret Service

Paul Resta
International Society for Technology and Education
Center for Technology and Education
University of New Mexico

Connie Stout
Education Program Director
Division of Technology Development
Texas Education Agency

Kathleen Hurley
Education Marketing
IBM Corporation

Sally Bowman
Director, Computer Learning Foundation
Palo Alto, California

Jim Mecklenburger
Director, Institute for the Transfer of Technology to Education
National School Board Association
Alexandria, Virginia

Special Presentation—ACM SIGCUE/ISTE

Technology Education for Teachers: Parts I & II—Focus, Activities, and Direction

Abstract

This session will introduce those in attendance to several special interest groups involved in technology education for teachers—both preservice and inservice. For each group, the focus, issues identified, activities, and direction will be briefly highlighted. The groups have evolved from different perspectives and membership backgrounds. It is important that the technology education community dealing with teacher education become cohesive and benefit from the activities and efforts of the others. Those in attendance will be involved in discussion. Unity of purpose and challenge will be explored with interaction of group representatives and attendees.

Panelists:

Henry Kepner, Jr.,
President, ISTE/SIGTE
Panel organizer and Moderator
University of Wisconsin
Milwaukee, Wisconsin

John Lawson
Chair, ACM/SIGCUE
Lewis Clark State College
Lewiston, Idaho

Jerry Willis
Coordinator, Annual Technology and Teacher Education Conference
East Carolina University
Greenville, North Carolina

Margaret Kelly
Chair, Association of State Technology-Using Educators
California State University
Fresno, California

Jackson Byars
Chair, Computer-Based Technology Education
Assoc. of Teacher Educators
Kansas State University
Manhattan, Kansas

Representative, American Association of Colleges for Teacher Education

Representative, AERA, SIG: Instructional Technology

Special Presentation-ISTE/SIGTC

The Role of Technology in the Restructuring of Schools

Abstract

This is a *Hypersession!* After brief opening remarks, the hyperpanelists, Alan November, Robert Pearlman, and Doris Ray, will serve as interactive, associative information bases for participants interested in exploring how technology can support and enhance restructuring schools. The participants will become part of the panel discussions, offering questions, comments, challenges, etc. The ensuing hyperlogue may also make use of videotapes, computerized images, or other audio-visual media.

Panelists:

Doris Ray
Director, Maine Computer Consortium
Auburn, Maine

Alan November
Glenbrook South High School
Glenbrook, Illinois

Robert Pearlman
AFT
Washington, D.C.

Projects

Hypermedia/Multimedia Applications

Animal Safari: Multimedia Learning for Special Needs Students

Christine Appert
Kluge Center
University of Virginia

Abstract

This presentation will provide an overview of Micro Mornings, an enrichment program for elementary-age children with learning disabilities and mild to moderate handicaps. Micro Mornings is sponsored by the University of Virginia Kluge Children's Rehabilitation Center and offers an intriguing sampling of organized small group activities during the summer and after school. Specifically, the Animal Safari project will be used to illustrate adapted computer-based experiences which take advantage of multimedia technology and integrated learning ventures.

The goals of the Micro Mornings program are to help students improve poor self-images, become more independent in learning tasks, take more responsibility for their own learning, and demonstrate the ability to cope with the learning process, including its inevitable times of frustration, confusion, and difficulty. Technology offers a vehicle for accomplishing these goals and provides a valuable medium for the flexible and individual planning required in special education. "Learning-centered" software, videodisc applications, and other interactive media provide a dynamic and meaningful context for learning which accomodates individual learning styles and abilities.

The Animal Safari project involved small groups, delineated by age, for six three hour sessions. Youngsters were introduced to various aspects of animal life through a variety of on and off-computer activities. Software selections includedpopular commercial programs,such as Sunburst's "Learn About Animals" or "Animal Trackers," Hartley's "Animal Survival," and National Geographic's "Project Zoo." In addition, *HyperCard* stacks were created to use with Videodiscovery's "Bio-Sci" videodisc and the "National Zoo" videodisc. Related projects encompassed a computer slide show with digitized images, creative writing, and charting. Sessions with the ASPCA or a localnature center complemented work with animal habitats and characteristics. Programs and activities were structured and adapted to adjust for variables such as deficits in reading and organizational skills.

Session participants will be given a description of the organization and special techniques employed in implementing this program. Specific suggestions for adapting software applications and creating program activities will be given. The unique benefits of multimedia for special populations will be highlighted throughout the presentation.

Hypermedia Instructional Enviroments: Learner Strategies and Learning Outcomes

Kim MacGregor
Louisiana State University, Baton Rouge

Abstract

Hypermedia, a rich instructional computing environment, offers possibilities for presenting information a format that allows for individual differences among learners. This project was undertaken to provide knowledge about secondary and college student's use of hypermedia. A hypermedia system was designed to present textual, graphical, and video information related to the subject area of interior design. A critical issue is whether students can navigate through cards and stacks purposefully without getting lost. Navigation through hypermedia systems requires that students demonstrate determines students' views of their control over the environment may be an important dimension of student success with this type of learning environment. formal measures of achievement and cognitive style provided information about student ability level and locus of control. Navigation through hypermedia may be influenced also by the use of instructional aids in the form of advance organizers. Two types of advance organizers (concept map or outline) were provided to students while they used the system. Information about the relationship between student characteristics, learner strategies, and learning outcomes will be presented.

White Plains Time Traveler—Local History on Interactive Videodisc

Kathy Rehfield-Pelles and Naomi Gross
White Plains Public Schools, New York

Abstract

During the Summer of 1990, four White Plains, NY educators developed an interactive videodisc computer project on the topic of Westchester history. Using resources and knowledge provided by fourth grade teachers who teach local history as part of their knowledge provided by fourth grade teachers who teach local history as part of there social studies curriculum, and with the assistance of Susan Swanson, Westchester County Historian and other local historians, the team compiled videos and text about the most important events and places in local history from the 1600's to 1940s. Over two hundred still or full-motion visuals were gathered from sites still standing or from photos and film clips in historical archives in the area. After puling together all the information, the team designed a Hyperstudio computer program for the Apple IIGS which allows users to view and interact with this history in non-linear ways. The events are indexed by time period and also by theme (Historical, Political, Economic, Geographic, Social) and can be accessed in any order. Users can follow events of a theme through different time periods or all of the events of a time through the different theme perspectives. This videodisc and computer program have become important resources for fourth grade teachers in there local history curriculum.

During the project presentation we will summarize the steps we took in designing and then producing our own videodisc and computer program. Emphasis will be place on ways that this project specifically takes advantage of the computer/videodisc medium (linking to next event to see, indexing swift and specific retrieval of visual data) and on the design elements we were careful to include that make this easy for the user to control and understand (mental mapping). We will demonstrate the completed project. We will discuss way in which fourth grade teachers are currently implementing the use of Time Traveler. We will briefly highlight other interactive video projects currently under development in White Plains.

Projects
Technology in the Elementary Setting

A New Direction: Distance Education and Chapter 1

Dianna Lawyer-Brook
RMC Research

Abstract

The Region 5 Rural Technical Assistance Center is currently piloting a distance education model using interactive television via fiber optic cable to deliver Chapter 1 instructional services. This experimental program is designed to provide math instruction to Chapter 1 students through the use of distance education technology at two locations, one in southeast Kansas and another in the panhandle of Oklahoma. Each location has a central sending site and three receiving classrooms. Students at a nearby school receiving the same instruction in a traditional classroom are used as a control group. The school districts provide the hardware equipment, while R-TAC is responsible for training, evaluating, and providing the curriculum.

The curriculum and methodology are aligned with current research, emphasizing the math process, problem solving, and the use of manipulatives/computers as mathematical tools. Teleteachers instruct the students five times per week on-line with paraproffessionals assisting in classroom management and homework projects.

Evaluation of this pilot focuses on outcomes form both the individual students and the program as a whole. Both formative and summative evaluation designs will be used to determine effectiveness. The results of this experimental program will prove useful to school districts as they make important decisions with respect to the use of technology in their schools.

Interpreting Literature Through Art and Technology: A Collaborative Project

Barbara Linsley
Mentor Teacher & Computer Trainer
Ontario-Montclair School District

Joan Hanor
Assistant Professor, Department of Education
University of LaVerne
Ontario, CA

Abstract

This project focuses on cross cultural interpretations of literature between Australian and Californian communities. It was implemented by two teachers, representing university and elementary levels, working together to establish creative channels for educational growth. Their purpose was to document ways technology could serve as a vehicle for examining, interpreting, and restructuring literature through visual analysis.

This pluralistic approach afforded students the opportunity to identify key elements of American Tall Tales, to express their reactions, ideas and feelings visually through art, and to communicate their analysis of related themes in collaborative groups using contemporary technological tools: AppleIIGS computers, video laser disks, Hyperstudio and 816 Paint.

Participation in WCCE '90 led to visitations to Australian Schools. This enabled first hand contacts in Australia for establishing the cultural exchange. Students there had shared their rich heritage of tales and an audio tape was provided for our students' benefit.

Current status of the project will be discussed along with goals, examples of learning activities, and samples of student work.

The WHAT'S IN THE NEWS Telecomputing Project

Chris Clark
Telecomputing Project Coordinator, WHAT'S IN THE NEWS
University Park, Pennsylvania

Abstract

Teachers and students in grades 4-6 dial the Learning Link network and receive up-to-date resources for a current events television series aired on public television stations. The goal is to enhance the effectiveness of educational television through telecomputing. Development began July, 1990, piloting in September. The service opened to Pennsylvania in January, 1991, and then nationally in March.

Telecomputing involves different levels of interaction. At one level, viewers receive program outlines, newspaper articles, library reference lists, thought questions, and schedule of programs. At another level, viewers communicate with program staff, students electronically submit news stories, and classes participate in monthly surveys. At the highest level, classes communicate with each other to conduct debates and carry out projects.

Celebrating its 25th season, WHAT'S IN THE NEWS is a nationally-distributed series. Its goals are to 1) put events and issues trends into perspective, 2) make news part of a student's life, and 3) strengthen current events education. Weekly 15-minute programs alternate between a summary of news events and a feature report on a single issue. Students become involved through essay-writing activities and "News Detective" segments. Thirty-one programs are produced annually by WPSX-TV in association with Penn State's College of Education.

Projects
Writing Across the Grade Levels

Implementing the Houston Independent School District Writing Labs

Dr. Richard Alan Smith
Director of Computer Curriculum
Houston Independent School District, Texas

Abstract

One of the most exciting developments in the Houston Independent School District is the recent establishment in each of our middle schools of computer laboratories dedicated to facilitating writing instruction. A gift of 1000 Deskpro computers by the Compaq Computer Foundation made possible the 38 labs, each with an average of 27 computers.

Our labs are unique because the students are taught to use the *writing process* to transform their thoughts into writing through six steps. The process begins with prewriting activities, writing a draft, and then discussing the draft; the students revise, edit, and finally evaluate the completed piece. The teachers were shown how to teach the writing process as they learned how to use the computers. Hence, rather than being simply an add-on to traditional instruction, the computers were seamlessly woven into a new state-of-the-art writing curriculum.

Accordingly, a visitor to any of the laboratories steps into an exciting environment where students are engaged in several activities. Teachers help students gather their ideas together, get those ideas down on paper, obtain suggestions for improvement from their fellow students, revise their papers on the basis of that feedback, edit their papers, and then evaluate their efforts. These activities are the key steps of the writing process. A frequent enhancement is publishing the papers in anthologies or newsletters.

Developing Creativity in Students through Multimedia

Stephen F. Bombara
Windham Public Schools, Connecticut

Abstract

Students are allowed to form groups of two or three, and given the task to design, manufacture, and market a real product.

Students will make their product known to the entire class to avoid any product duplications. They will have to manufacture a sample, and create a package for this product.

Students will name their company, develop a slogan, and draw a logo to be used for their letterhead. Students will learn to use the scanner to scan their drawn images.

During one class students will view favorite TV commercials and discuss the elements present, which make a commercial successful.

Students will learn how to use a word processor and desktop publishing package by drafting:

- A brief history of their company (how and why they got started).
- Give detailed information about their product (eg. recipe, ingredients, instructions).
- Develop a letterhead for their company.

Students will learn about databases and develop one by researching companies, that might buy their product, from a phone book.

Students will write to these companies asking for a time and place to meet for a sales presentation.

Prior to the meeting students will create marketing materials. These materials will include: transparencies of charts and graphs, created on a spreadsheet .

Students will write and act out a script for a radio commercial which will be aired over the school intercom.

The final task is to draft a script, make props, and design a set for the taping of a video commercial. These commercials will be shown at school based functions.

An Interactive Writing Program for Urban Teenagers

Randall Ryder
Milwaukie, Wisconsin

Abstract

Abstract not available at press time.

Special Presentation

Information Literacy: A Changing Education Environment

Abstract

Information is exploding using imagery, sound, and movement to influence and create impact. Text literacy is only one of many mediums for communication. Print, video, graphics, and sound each take very different approaches to reaching the mind. The impact of different information modes on thinking and learning will be demonstrated. Information presented will be research and experience based. Students can move from passive text information learning to knowledge building with dynamic interactive tools. Are your students going to be literate beyond print for reading and writing information?

Panelists:

Bernajean Porter
Senior Consultant
The Confluence International Group
Boulder, Colorado

Ron Fortunato
Senior Consultant
The Confluence International Group
Virginia Beach, Virginia

Special Presentation—ACM/SIGCSE

Computer Science: A College Preparatory Course for All High School Students

Abstract

In 1983 a task force working under the ACM Executive Board's Elementary and Secondary Schools Subcommittee drafted recommendations for courses in Computer Science at the secondary school level. These recommendations were published in 1985, jointly approved by ACM's Education Board and the IEEE Computer Society's Education Committee. Many changes have taken place since that committee completed its work. Among them are the report of the ACM Task Force on the Core of Computer Science and a curriculum model for an introductory course. The ACM Pre-College Committee have organized the task of developing a corresponding secondary school course. The proposed course is a one year, college preparatory upper division high school course for all students, comparable in scope and rigor to general biology, chemistry, or physics. Panelists will propose course models and solicit discussion.

Panelists:

Susan Merritt, Moderator
Chairperson, ACM Pre-College Committee and Dean
School of Computer Science and Information Systems
Pace University
Pleasantville, NY

Philip East
Associate Professor and Acting Head, Department of Mathematics and Computer Science
University of Northern Iowa

Darlene Grantham
Secondary Teacher Specialist
Montgomery County Public Schools
Maryland

Charles Rice
Mathematics Department and Computer Coordinator
The Dalton School
N.Y., N.Y.

Carol Wolf
Chairperson, Department of Computer Science
Pace University
N.Y., N.Y.

Lesley Klein
IS Department
Panel Coordinator
Pace University
Pleasantville, N.Y.

Projects

Using Computers to Collect, Analyze, and Synthesize

Hands on Data: Putting Kids in Charge of Data!

Phyllis Kalowaki
Newton, MA

Abstract

The goal of the Hands On Data project, supported jointly by TERC and the Cambridge School Department, was to develop interdisciplinary units in which students formulate a problem, gather and interpret data, and communicate their conclusions. This process was aided by use of prototype software, the Tabletop, which helped translate NCTM standard regarding data analysis into lively classroom activities.

This session will provide a practical review of a year long research project involving a fifth/sixth grade and an eighth grade class. By peeking into the classrooms, the results of using technology along with inquiry learning activities will be discussed and the impact examined. We will analyze how teachers, In both a self-contained and departmentalized situation, went about developing these units. Some issues arose regarding curricular integration, integration of technology, classroom management, and teacher support. The efforts and progress in resolution of these issues will be discussed.

The Tabletop: A Visual Tool for Creating, Exploring, and Analyzing Data

Chris Hancock
Project Director, Hands on Data
Technical Education Research Centers, Inc. (TERC)

Abstract

The Hands On Data project has been developing and piloting a data exploration and analysis tool called *The Tabletop*, as an exemplar of a new class of tools for working with data. The Tabletop is a highly visual, highly interactive interface that responds to queries visually through animated icons on the screen. the software has the ability to arrange these icons into Venn diagrams, graphs and scatter plots according to attributes determined by the user. This visual representation allows concepts ordinarily considered "advanced" to be understood at a surprisingly early age. The software's quick feedback encourages students to explore and to experiment with their own ideas.

This presentation will demonstrate the Tabletop, mention some principles underlying its design, and discuss its intelligibility to students of various ages. It will also consider briefly what kinds of skills and concepts underly the use of computerized data tools and the new kinds of mathematics, and new meaning of the term "computer literacy," which these skills and concepts suggest.

The Hands On Data project is conducted by TERC in collaboration with southeastern Massachusetts University. It is funded by the National Science Foundation with additional support from Apple Computer, Inc.

Exploratory Data Analysis, Median Polish, and Mathematica

Martin Adickes and Rob Rucker
Industrial and Management Systems Engineering and Center for Computer Integrated Manufacturing
Arizona State University, Phoenix

Abstract

The work presented here grows out a need to teach undergraduate engineering students approaches to analyzing real world data. These approaches are designed to help them in their initial investigations of data sets, with a view toward detecting unusual or important features prior to formal hypothesis testing. Data from real environments does not usually lend itself to easy application of standard confirmatory statistical analysis techniques and so other, complementary, approaches are often necessary. One such approach is that represented by the field of Exploratory Data Analysis (EDA). Within that field we have chosen a particular technique, Median Polish, to introduce to our students. The first part of this paper explains median polish and associated techniques, in an introductory manner. The last part of the paper describes the mathematical environment available within the computer package *Mathematica*. An appendix presents the outline of our interactive tutorial on this topic. We have written the tutorial in a 'notebook' style using the programming tools within *Mathematica*, an environment for doing mathematics. This computer environment, together with accompanying text material, allows the student to experiment with the concepts and the consequences of analyzing data in an exploratory manner. The material has been presented to students who had taken an introductory statistics course and were 'Macintosh literate.' The results were positive on the whole but pointed out the need for supplementary instruction in more basic skills such as finding medians and matrix operations in general.

Projects
"Take a Risk" Students with Special Needs

Enhancing Learning through the Use of Interactive Video

Doris Tyler
Chapel Hill High School, North Carolina

Abstract

The Sophomore Seminar: Program for Academic Challenge currently involves 85 students and is designed to enhance student's ability tostructure, analyze and communicate information across the content areas of social sciences, science, mathematics, and English. Through an interdisciplinary, project-oriented curriculum, we are helping students to see themselves as successful learnersand capable young adults. A four-hour block of uninterrupted time facilitates a true interdisciplinary approach, helping students understand the connection of these disciplines with the real world.

Two major focuses of all seminar activities are to improve student self-esteem and to enhance skills related to accessing, processing and communicating information. Students were placed in cooperative learning groups to research and develop a multimedia presentation for biology. Traditional research skills as well as on-line databases were used to locate information in the media center.

Students were taught to use the Macintosh computer, *HyperCard*, and a videodisc player. Student created interactive video stacks were created using the Life Science videodisc from Optical Data. Each research presentation included information about one phylum or class of organisms; the information presented depended upon the research as well as the group of organisms but included the general body structure, the geographic habitat, place in the food web, and usefulness to humankind.

Mac-in-Touch

Diane Frank, Carol Goltz, Mary Margaret Stephens
West Anchorage High School, Alaska

Abstract

The three presenters are completing the second year of an Apple "Equal Time" Grant which supports an interdisciplinary project targeting at-risk ninth graders and which emphasizes the development of *HyperCard* stacks and interactive media. Enthusiasm for *HyperCard* has put the Mac in touch with kids and curriculum throughout the school. This presentation will show the evolution of interdisciplinary HyperCard stacks developed by students ranging from at-risk freshmen to advanced placement senior, demonstrating stack development from the simple to the sophisticated as students become more comfortable and creative with the technology.

From Desktop Publishing to Hypermedia

Kathleen Sutphen
T.E. Mathews County Community School
Marysville, CA

Abstract

T.E. Mathews County Community School, located in Marysville, California, was established to serve at-risk youth. Our 45 students range from grades 6 through 12. The majority of our students have low academic skills, low self-esteem, and little or no history of success within a school environment.

Against this setting we were awarded and "Equal Time Grant" by the Apple Computer Corporation for the two consecutive school years, 1989 through 1991. The focus of our grant was to have our students publish a community newsletter, *The Other Side*, using a fully networked Macintosh computer lab donated by Apple. Our major software program for publishing is Aldus *PageMaker*. Our first year was an unheralded success.

During the 1990-91 school year we combined a California Technology Grant with our Apple Equal Time Grant. The focus of our new grant was to augment our existing technology use with use of hypermedia. With some new equipment and software, our students began to create a *HyperCard* based software program that gives the potential user an interactive medium to learn about the history of our culturally rich geographical area. We produced our newsletter in conjunction with development of our software program, using researched topics, interviews, pictures, recordings, etc., for both products.

We have come a long way over the past two school years. The evolution of technology has been fast-paced and we and our students have run a successful footrace keeping up with it. We are all enthusiastic. Our presentation will highlight what we've learned, "tricks" that helped make it work, and a sampling of our students' hard work.

Projects
Teacher Education Telecommunications

Using PROLOG and Hypertext in Undergraduate Teacher Preparation

Gary Tubb
Louisiana Tech University

Abstract

The presentation will focus upon documented uses of PROLOG and hypertext in an undergraduate course that integrates computer education with media. Techniques for teaching LOGO in the course are contrasted with teaching PROLOG. Emphasis is placed upon graphics and database notions that each language conveys. Experience with the construction of a simple expert system for media selection, as part of the course, will be shared in the presentation. Finally, experience in using a simple text-based hypertext system in the course will be shared in the presentation.

Education Students On-Line at Cal State Los Angeles

James H. Wiebe
California State University, Los Angeles

Abstract

This presentation will describe how the Instructional Technology program at California State University, Los Angeles began its own BBS; how the BBS is maintained and run; how it is used in introductory courses for teachers and in our graduate programs; and our plans for it.

Our BBS was started two years ago on an IBM-XT clone with a 20 megabyte hard drive and using a public domain BBS program, *Opus*, Version 1.03. We are a node on FIDONet, a national network which allows our students and faculty to send electronic mail, access various forums, obtain public-domain software, etc. Our BBS has been enhanced through several small grants, including one which allowed us to purchase 15 modems to lend to students so that they could use the system from home and another which allowed us to develop a variety of curriculum materials to use in introductory computer courses for teachers. We recently logged our 5000th call. Fifty-five percent of these calls have been course-related.

Our enhancements to the system have dramatically increased usage of and user satisfaction with the system. A recent Instructional Technology MA project evaluated usage of the system. It was found that there has been steady growth in use and that the system has enhanced faculty-student interaction outside of class time.

Wednesday

NECC '91 CONFERENCE PROCEEDINGS

Special Presentation

Education on the Electronic Frontier: Learning and Teaching on Tomorrow's Networks

Developments in computing and telecommunications have made possible new forms of learning and teaching. This panel will address a number of questions raised by these developments. It is now possible for students and teachers to interact with adults outside the school system. What educational models will support this kind of interaction?

With access to expertise and information outside the classroom, teachers and students can become true inquirers. What changes will be necessary for such inquiry to take place on a large scale in education? What sources of expertise and information should teachers and students be able to access? How can students and teachers be taught how to locate what they need when they need it? How do kids and teachers learn to make sense of all the information they get, once they tap into it? What incentives are there for experts to provide services to kids and teachers "on demand"? This panel will address these questions from several different perspectives, including those of K-12 schools, higher education, government and business.

Panelists:

James Levin
University of Illinois

Michael Waugh
University of Illinois

Beverly Hunter
National Science Foundation

Margaret Riel
AT&T Learning Network

Al Rogers
FrEdMail Foundation

Projects
Project Outcomes

Computers and Classroom Instructional Research

Shirley Becnel
Jefferson Parish Public Schools, Louisiana

Abstract

For the past three years, the Chapter 2 department of the JPPSS has funded more than 20 research based instructional projects for the purpose of fieldtesting and evaluating innovative programs in the classrooms prior to recommending district wide changes in the instructional program. Six of the completed, successful, projects used computer technology as an essential part of the instructional design. As a result of the fieldtests, three of the projects (1,2,3) have been expanded and incorporated into district curriculum. Three other projects (4.5.6) yielded data which were used in the restructuring of specific subject area guides.

The purpose of this presentation is to report the results of six technology based research projects. These projects are: 1) a district wide high school writing project which addresses both the general writing/editing skills of secondary students and the specific skills needed to pass the writing portion of the state high school competency test; 2) a district wide science technology project using computer probes and sensors in biology, earth science, and physical science classes; 3) a high school biology project using videomicroscopy; 4) two computer based writing projects, one involving middle school special education students and one for handicapped primary level special education students; 5) an algebraic graphing project utilizing large screen computer display units and graphing software in pre-algebra classes at the middle school level; and 6) a computer based reading vocabulary project for over-aged, academically at-risk, non-special education elementary students in grades 5/6. Note: Project 1 was implemented at 7 high schools and project 2 involved students in 8 high schools and 17 middle/junior high schools. Other projects involved one or more classes at a single school site.

Participants will receive a description of each project including computer and equipment specifications, copies of the project designs, evaluation data, and the final reports.

Technology Now and Tomorrow—An Interactive Approach to Elementary Science

Beth Sandez
North Ridge Magnet School
Moreno Valley, California

Abstract

In a time when the drop-out rate is rising and scores on standardized tests are declining, the question for educators is: How can we equip students to be independent lifelong learners capable of creating and utilizing tools to solve problems and improve life? Since today's elementary school aged children were born into the world of Nintendo, VCR's, and electronic games as toys, the answer seems to lie in the area of interactive technology. Paper and pencil are simply not sufficient tools for today's technology aware students.

Schools are gradually learning from business and industry to use technology as tools to instruct and motivate students. One of the most exciting uses of technology in the elementary school is interactive laser disc programs, especially in the area of SCIENCE. There are wonderful laser disc programs which bring the world of space, nature, or earth science right inside the classroom! Come to see what a Science, Math, Technology Magnet elementary school is doing to prepare its students for the twenty-first century!

Indiana's Project 3R's

Julie Poage
Ball State University, Indiana

Janelle Zorich
Apple Computer Inc.,
Indianapolis, Indiana

Abstract

On June 13, 1990 Indiana's Governor Evan Bayh and the Superintendent of Public Instruction, H. Dean Evans, announced the implementation of **Project 3R's**, a program designed to extend educational technologies in elementary schools throughout Indiana . The program made available $2,000,000 to aid in the establishment of educational technology laboratories for instruction in elementary reading, writing, and mathematics, focusing primarily on kindergarten and/or first grade programs.

The project entailed the completion of three components: (1) petition and slection of participating school corporations; (2) the training and support, including hardware, software, and curriculum development; and (3) evaluation of the project through the maintanence of student achievement indicators.

The Indiana Department of Education evaluated school corporation petitions on the basis of how well they addressed the intent of the enabling legislation (HB 1290, Chapter 25, Sections 1-4) and the extent to which the implementaion plans addressed specific guidelines, such as the commitment of matching funds and staff support.

The training and technical support associated with the use of Apple equipment included regional technical support, training sessions with *Apple Learning Series: Early Language Training* , and curricular integration specific to language arts, writing, math, and problem solving provided by staff at Ball State University. Follow-up training sessions have been scheduled at regular intervals throughout the 1990-91 academic year and will address specific needs expressed by participating teachers.

Evaluation of **Project 3R's** will include pre and post survey data concerning identified needs and staff development expectations of project participants, as well as descriptive statistics of student achievement gains. Qualitative data will also be collected from randomly sampled school corporations, including the level of impact and satisfaction with **Project 3R's**.

Special Presentation
Visual Arts —A New Frontier in Computer Technology

Nancy Scali
Ontario-Montclair School District
Ontario, California

Visual Arts features an Arts Education Technology presentation developed for the J. Paul Getty Trust for Education in the Arts. Emphasis will be placed on the creative and technological processes used in the development of a child or adult's computer artwork. The communication, artistic expression and pictorial interpretation of creative visual thinking will also be explored. A mini gallery of professional and student computer artwork will be on display. Discipline Based Art, Fine Art and curriculum integration via Art will be shared.

Fine Arts projects featured in *The Writing Notebook* and Sunburst's *Solutions* will be on display.

Projects

Computers in the Future

Imperative: Technology Training for Administrators

Hilary Cowan
Kanawha County Schools, West Virginia

Abstract

The Effective Schools research emphasizes the crucial importance of strong and purposeful leadership by the principal. At the same time, advocates of restructuring education have emphasized the need for effective use of technology to increase student productivity. Despite the general agreement on these issues by national educators, almost no attention has been given to the training that will be needed by school principals and central office administrators to understand their role in achieving technological excellence in their schools.

If we are to assume that administrators need training to become effective technological leaders, then we must address the question of how to achieve that goal. This includes an understanding of what components need to be developed in order to teach administrators what they need to know. It also needs to include a reality assessment ot the time and effort that can be expected from administrators who are already under pressure to implement other major insructional strategies. Furthermore, one cannot ignore the all too common fear by administrators that their ignorance of technology will be exposed and will impact their role as a leader.

For these reasons, training must be developed around three concepts: 1. limiting the time and number of sessions while maximizing learning, 2. limiting the training classes to peers, so that administrators can feel free to make mistakes and ask important questions, and 3. focusing the training on effective *supervision* of technology rather than on effective *use* of technology.

Although aspects of the training will be identical with that which is developed for teachers, certain issues are unique and others must be expanded or reduced to fit the administrative needs. It is important, for example, that principals understand their unique role in establishing a consistency in the school use of technology. It is also essential that they learn how to evaluate the use of technology by teachers in classrooms. On the other hand, although some hands-on time with software is necessary, administrators do do not necessarily need to learn how to use any program in great detail. It is far more important for them to be able to assess whether the program itslef is consistent with county, school, and classroom instructional goals.

The above issues are a few of those which will be covered in more depth in the presentation. Practical considerations, such as the politics of maximizing participation in training will also be discussed.

If technology is to move forward in schools across the nation, then we no longer have the option of simply training our teachers andand hoping that consistency of purpose and excellence of technological use will filter through our classrooms. The time for emphasizing the need for administrative training is now.

The Florida Science Videodisc Project

Robert Fronk

This is an update on the 1991 Florida Science Videodisc Project. The initial 1990 project has received further funding and is being continued into 1991. This year 30 Florida elementary and middle/junior high school teachers will get together at Florida Institute of Technology in Melbourne to learn about and work with computers and videodiscs. In addition to receiving training in interactive video development in the project's educational technology laboratories, participants will receive a videodisc player, a stipend, room and board for those staying on campus, videodiscs, computer software, graduate credit, and other materials. The project will create the second "Florida Science Videodisc" with slides and video from the 30 teachers and F.I.T. faculty. The 1991 project will focus on the life sciences (last year's work dealt mainly with the physical sciences). The 1991 Florida Science Videodisc will include all of the nearly 2000 physical science images from last year's project in addition to approximately 2000 new life science slides from this year's project. As long as the supply lasts, the 1990 Florida Science Videodisc (with its associated software for Apple II, Macintosh, and MS-DOS) may be purchased for $20.

The School of the Future: Visions for Learning

Gwen Solomon
The School of the Future, New York

Abstract

The future is now! New York City's School of the Future, a public secondary school, opened in September, 1990 in midtown Manhattan. This is our vision of education for the future: the methods and technologies we use to help our students learn 21st century skills.

We hope to help students get, understand, and manipulate information, learn how to learn, think creatively, make decisions, solve problems, and work collaboratively.

Our students are engaged in active learning; they create presentations and performances. They collaborate on projects to pool strengths and learn from one another. We have creative scheduling so students spend longer periods of time working in depth at their tasks. We have an interdisciplinary Humanities (English and social studies) curriculum that is based on a thematic approach to American history. We find that technology provides wonderful and important tools for student learning.

Many elements help us achieve success as we create a learning environment for the 21st century.

Projects
Mathematics Applications: K-University

Ease Students into programming with Turbo Hal

John T. Taylor
Hillsborough Community College
Tampa, FL

Abstract

TURBO HAL: the Integrated Environment, has been developed as a tool introducing students to structured programming, especially in beginning Pascal classes. It is designed to be a low cost alternative programming simulator to Karel, the Robot—a well-known such simulator. It is an easy-to-use integrated environment that does not require a seperate editor, compiler, linker, and debugger in order to create, debug and run Karel programs. HAL lives in the Borland language environment with its editor, windows and pull-down menus. It thus eases the student into a popular programming environment. This software was designed by Tim Totten of Hillsborough Community College

Turbo Hal has been successfully used with elementary and junior high students. Initially, a lifesize three dimensional grid made from a bed sheet and felt strips is used as a "playing board". The students are given programming template handouts on which they write-in the primitive commands to solve a typical HAL problem in the 3-D world. A student then walks through the grid as though he/she was HAL following the student's program. While observing "HAL" walking through the grid, the other students eagerly check and "debug" their own programs to solve the problem.

Since HAL can only make a limited number of simple moves. constructing more complicated moves, the students must use, and thus learn, the concept of a procedure. Older elementary and junior high students are easily introduced to iteration and conditionals as well as loop structures. By using a combination of the 3-D world and HAL's 2-D world on the computer, students are successful programming solutions to more complex mazes.

Teachers as well as students may generate their own problems by using the HAL'S WORLD generator or watch HAL execute step-by-step instructions solving many different HAL problems

I have designed a hypertext environment for HAL, using IBM's LInkway. It includes sample tmeplates for the beginners and instructions on how-to-build HAL grids. From any "page" (screen), the user may access a lesson or slide show to review or reinforce the programming concepts. Turbo Hal may also be accessed at anytime from any screen.

Turbo Hal runs on low-end MS-DOS machines and operates in text mode so that no graphics card is needed. It is available on a 5.25" disk. the Linkway version requires a mouse, CGA graphics and 640K RAM. It is available on a 3.5" DD disk. All software is distributed by Hillsborough community College.

All participants will receive copies of Turbo Hal for their school's use. a runtime version of the linkway hypertext presentation is available for the price of an exchange disk.

Using Hypertalk Scripting to Introduce Beginning Algebra Concepts

Barbara Dubitsky
Bank Street College

Rita Kabasakalian
New York City Lab for Gifted Education

Abstract

We have found that children learn mathematics more easily and with deeper understanding when they are engaged in solving mathematical problems that interest them. Certain kinds of computer programming activities provide interesting problems to enrich the mathematics classroom. We had been using the Logo language to provide these experiences, but this year we began using Hypertalk.

Seventh and eighth grade students in The Lab School, a public junior high school in New York City were taught Hypertalk scripting as part of their regular mathematics classes. The varied uses of variables as well as the use of graphics with a coordinate system were emphasized.

The concept of the variable was easily understood as the students explored the commands "ask" and "answer" and the uses of local variables with both words and numbers. Some children needed the "repeat" and "repeat with" commands for their work; several of these children easily understood recursion, others could use it only mechanically. We introduced graphics and animation possibilities within scripts, these required extensive use of a coordinate system. We found the rotation possibilities to be of particular interest to the students.

When we then used these concepts in more traditional mathematics work, we and the students referred often to the Hypertalk experiences. We found that all the students could easily switch from the Hypertalk coordinate system to the usual Cartesian system and that they understood the use of variables in their algebra work.

MathTech
A Collaborative Project of PLAYING TO WIN and COMMUNITY SCHOOL DISTRICT #4 (NYC)

Andrea Kimmich-Keyser (Playing to Win)
Gayle Rubenstein (District #4, New York City)

Abstract

MATHTECH was planned as a collaborative project involving parents, children, and their teachers in exploring technology-using extensions of the children's math curriculum. Teachers from fifteen elementary and intermediate schools, each with a mathlab equipped with an Apple IIe computer, participated. students and their parents attended weekly out-of-school computer sessions at the Playing to Win Computer Center.

In a typical MATHTECH project, and following in-class use of Math Shop (Scholastic software), students and their parents surveyed local stores, interviewed proprietors, collecting examples of real life use of mathematics, and developed a neighborhood directory of stores along with a descriptive map showing their locations. In the process, both children and parents learned to use word processing to write up the interviews, dat bases and spreadsheets to record survey results, and computer graphics to create the map.

This activity, along with other MATHTECH activities, will be described in a manual for teachers in other communities who wish to promote parent child collaborative math learning.

The presentation will describe some of these other project activities, show samples of student work, and report on results obtained.

Paper

K-12 Computer Teachers Literacy Status and Training Assessment in Illinois

Luann Stemler
Joaquin Vila
Illinois State University

Abstract

The purpose of this study is to report the current literacy status and training needs of K-12 computer science teachers in the State of Illinois. In order to gather comprehensive K-12 information on computer teachers, a survey was sent to those individuals whose major teaching assignment was computer related in public school systems. The total population of such group was surveyed (Illinois State Board of Education Data Base). Study findings, based on a 54% response rate, included the following: (1) Only 9.4% of the respondents hold a degree in computer science; (2) 85% have taught BASIC programming at some time in their career, but only 6.5% of those teachers have formally studied the language; (3) Over 95% said they are using microcomputers in the classroom as opposed to minicomputers or mainframes; (4) More than half of the K-12 computer teachers have interest in attending graduate courses or workshops to learn about multimedia technology in the classroom.

Introduction

The computer is now a familiar sight in most schools. Taylor and Poirot (1984) stated that the movement of computers into the school system has presented a problem in the lack of teachers trained in the use of computers as an aid to the educational process and a lack of teachers who can teach about computers. While computer utilization continues to expand in the schools, the training of teachers to make use of the computer and to teach the topic of computing has significantly lagged behind.

Poirot, Taylor and Norris (1988) survey the state computer educator coordinators and found that most states are able to find faculty to teach high school computer science. The coordinators expressed doubt about finding quality faculty to offer the more advanced courses anticipated in the future, particularly when computer science certification standards are enforced. The conclusion was drawn that retraining is critical if computer science is to develop as a discipline in the high school.

The ACM Task Force on the Core of Computer Science anticipates and demands high school courses provide a computer science background far beyond the elementary level courses commonly found in the high school today. Cooperation between universities, state education coordinators and districts are needed to develop and implement programs that provide for the special needs of the local teachers. The computer teacher needs to become a true computing professional who can adapt to a changing, expanding curriculum.

The study was undertaken in order to better understand the existing status and training needs of the K-12 computer science teacher relative to the teaching of computer science in the public school systems of Illinois. The survey also grew from a desire in the Applied Computer Department at Illinois State University to develop variety of courses relevant for the K-12 computer science educator. Casual observation had led to the conclusion that the present quality of instruction covered a wide range, with some instructors doing a fine job in the classroom, while there are some instructors who sorely need to upgrade their computer skills.

Research Methodology

Population

The survey was mailed to K-12 computer teachers in public school systems in the state of Illinois. The population source came from the Teacher Service Record collected by the Research and Evaluation Department of the Illinois State Board of Education. All individuals whose major teaching assignments was categorized as data processing, computer literacy/awareness, or computer programming were surveyed. Surveys were sent to 627 computer educators and received 340 responses, a 54% response ratio.

Survey Instrument

A survey questionnaire was constructed that consisted of items assessing four areas of interest: (a) teacher background, (b) computer literacy assessment, (c) computer hardware (d) training interest.

- The teacher background component surveyed major area of concentration and degree held (BS/BA, MS/MA and PhD.)
- The computer literacy component assessed credit computer science hours taken. Those hours were further analyzed by programming languages, Data bases, literacy courses, etc.

- The computer hardware component inquired about existing computer resources available at their school.
- The training interest component explored the current and future areas of interest of the computer science teacher. An extensive list of possible courses were presented including: multimedia technology, Computer Aided Instruction, Teaching Strategies for programming instruction, etc.

Findings

The respondents in this study reported only 9.4% of the computer science teachers in the State of Illinois hold a degree in computer science. Business, Mathematics, and Elementary Education are the predominant disciplines. (See Table I)

Only 9.1% of the responding computer teachers have taken less than 5 hours of credited computer science courses. Whereas, 47.9% have taken over 16 hours. (See Table II)

Table III displays the credited courses taken by the target population. 74.7% of the respondents took Introduction to Computers and 77.4% have taken some programming course.

Table IV shows the programming languages that are being taught in the K-12 school system, and further shows the interest areas of the classroom teacher. 85% of the computer teachers have taught BASIC programming at some time in their career. Surprisingly however, only 6.5% of those teachers have formally studied the language. Furthermore, the questionnaire indicates that 30.3% of the respondents are interested in learning the C programming language followed by Pascal.

The survey revealed there are several ways the teachers are coping with the ever changing field of computer science. 36% of the respondents have attended graduate courses. 64% have attended computer related workshops. Table V shows how computer teachers are learning on their own.

Table I
Major Areas of Concentration

	BS/BA	*MS/MA*	*PhD*	*Total*
Accounting	4.1%	0.0%	0.0%	4.1%
Business	24.1%	15.6%	0.6%	40.3%
Computer Science	3.5%	5.3%	0.6%	9.4%
Economics	0.9%	0.6%	0.0%	1.5%
English	2.9%	0.9%	0.3%	4.1%
Elementary Education	13.2%	9.1%	0.0%	22.4%
Foreign Languages	0.6%	0.3%	0.0%	0.9%
History	1.5%	0.3%	0.0%	1.8%
Home Economics	0.3%	0.3%	0.0%	0.6%
Industrial Technology	0.3%	0.9%	0.3%	1.5%
Mathematics	23.8%	11.2%	0.3%	35.3%
Science	5.3%	2.6%	0.0%	7.9%
Other	0.6%	7.9%	0.6%	9.1%

Table II
Computer Science Credited Hours

0 - 4 hours	9.1%
5 - 8 hours	9.4%
9 - 12 hours	17.4%
13 - 16 hours	12.4%
more than 16	47.9%

Table III
Credited Courses Taken

Introduction to Computers	74.7%
Word Processing	40.3%
Data Base	30.0%
Spreadsheets	29.1%
Desktop Publishing	9.4%
Microcomputers in the School	42.6%
Programming	77.4%
LOGO	21.1%

Table IV
Programming Language Course Status

	Courses Taught	*Courses Taken*	*Courses Interested in Learning*	*No Interest*
BASIC	85.0%	6.5%	2.4%	2.6%
Pascal	19.7%	23.2%	19.7%	16.5%
COBOL	21.2%	19.1%	10.6%	22.6%
Assembler	5.9%	20.6%	12.1%	27.1%
C	1.5%	2.1%	30.3%	24.1%
PL/I	1.2%	9.4%	10.6%	32.9%

Table V
Self-Instruction

Reading computer related articles	78.5%
Reviewing new software packages	75.6%
Any combination of the above "all by themselves"	67.9%

Most K-12 computer teachers responded that they are using the microcomputers (95.5%) as opposed of minicomputers or mainframes. Figure 1 depicts the ratio of microcomputer type used.

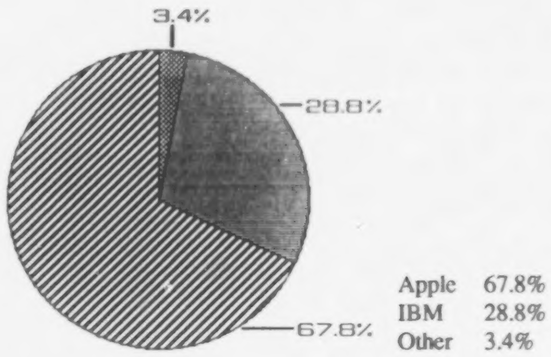


Figure 1. Microcomputer Type Used

According to the survey, there is broad interest in attending graduate courses and workshops. Table VI shows distinct areas of interest.

Table VI
Computer Area Interest

Introduction to computers	4.1%
Programming Courses	22.0%
Structured Programming	15.6%
Teaching strategies for programming instruction	39.1%
Hardware \ Computer Architect	22.4%
Software Design	25.0%
Using Multimedia Techniques in the classroom	51.2%
Artificial Intelligence	31.5%
Computer Aided Instruction:	37.1%
Authoring Software	22.4%
Gradebook	17.1%
Wordprocessing	22.0%
Spreadsheets	25.0%
Database	24.7%
Desktop Publishing	45.9%

Conclusion

Overall, the survey proved very fruitful. Information collected on the K-12 Computer Teacher Questionnaires satisfied the objectives formulated for the survey. The K-12 computer teachers current literacy status and training needs were assessed. The outcomes of the survey indicated that formal training is wanted to strengthen concepts taught at K-12 level as well as to facilitate the instructor mission through proper professional development.

Survey results were also useful in identifying and ranking the topical areas for (credit and non-credit) workshops and graduate courses. The training needs emphasized by computer teachers were in the areas of multimedia technology, strategies for programming instruction, computer-aided instruction, and artificial intelligence.

References

Denning, P.J., Commer, D.E., Gries, D., Mulder, M.C. Tucker, A., Turner, A.J., & Young, P.R. *Draft Report of the ACM Task Force on the Core of Computer Science*, February 1988.

Poirot, J.L., Taylor, H.G., & Norris, C.A. "Retraining Teachers to Teach High School Computer Science. *Communications of the ACM*, 31, pp. 912-917, 1988.

Taylor, H.G., & Poirot, J.L. "Computer Science Teacher Certification: Current Status and Trends." *T.H.E. Journal*, pp. 103-107, 1988.

Paper

Experiences Retraining Elementary and Middle School Teachers To Teach Computer Programming

Judith D. Wilson and Robert M. Aiken
Computer and Information Science Department

Glenn E. Snelbecker and Nina P. Bhote
Educational Psychology Department
Temple University

Abstract

A project implementing a model retraining program in computing and computer technology for elementary and middle school teachers is described and assessed. The project, still in process, is retraining teachers to become teachers of computing and computer resource people for their schools. As computing teachers, these teachers will be expected to teach computer programming, and programming is normally taught at the elementary and middle school levels with the objective of developing mathematical, problem solving, and learning skills. Although the project has been very successful as judged on several objective measures, informal observations suggest that the courses that teach programming may need to be modified to reflect changing views about the needs of teaching computing at the elementary and middle school levels.

1. Introduction.

Many educators would agree that elementary and middle school students can benefit from using computing tools to support knowledge acquisition and problem solving activities throughout the curriculum. Some envision a revised curriculum where computing plays an integral and interdisciplinary role (Burke, 1990; Goodspeed, 1989; Haney, 1990). Many would also agree that the ability to program a computer is an important skill to have in a computer-based information society. Additionally, although the supporting evidence appears inconclusive, many educators believe that learning to program a computer can have beneficial side effects for students at all grade levels, including reduced apprehension about mathematics and development of general learning and problem solving skills (Papert, 1980; Pea, 1989).

Educational software tools have been designed to facilitate learning in many subject areas in the K-9 curriculum, including mathematics, science, and language. Numerous software packages are now available and new packages enter the educational market frequently. To benefit from evolving computer technology, schools need teachers who are able to evaluate and select hardware and software to support educational objectives, integrate computer resources into the curriculum, and train teachers to use new technology. Teachers trained to conduct such activities are known as computer resource people or "teacher technologists".

We distinguish the teacher technologist from the computing teacher. Computers are the focus of courses in computing which may include introductions to computers and applications packages but which generally teach computer programming using either Logo or BASIC. In elementary and middle schools, programming instruction usually is aimed at fostering mathematical, problem solving, and learning skills. Thus elementary and middle schools need teachers prepared to function both as teachers of computing, teaching problem solving and learning skills through programming, and as teacher technologists.

Considerations such as these have influenced the design of a model program to retrain experienced elementary and middle school teachers to become teachers of computing as well as teacher technologists. We describe and appraise an implementation of the model which is funded through a National Science Foundation project grant. Now that the project is more than two thirds completed, it is possible to make a preliminary assessment of it as a vehicle for retraining teachers with little or no computing experience to teach computing. Although the project has been very successful generally in achieving its objectives, there are some early indications that those aspects of the model program concerned with retraining teachers to teach programming may need to be modified.

2. A Model Retraining Program for Elementary and Middle School Teachers.

The retraining program, which is still ongoing, is being conducted by Educational Psychology and Computer Science faculty at Temple University in cooperation with the Philadelphia School District. Teachers from the school district were selected to participate in the project on the basis of motivation and likelihood of success as indicated by past achievement. Previous computing experience was not used as a selection criterion, and care was taken to select participants from a variety of different backgrounds.

2.1 Curriculum Goals and Structure.

The goal of the model retraining program for elementary and middle school teachers is to prepare teachers to perform the broad array of functions expected of computing teachers in elementary and middle schools. Thus the general goal of the project may be viewed in terms of several component goals: (1) preparing teachers to become teacher technologists, (2) training teachers to teach computing, and (3) focusing on teaching strategies for computing courses that foster the development of general problem solving and critical thinking skills. Measures of success in the attainment of these goals include having more than 50 percent of the teachers complete the program, and having graduates of the program pass a comprehensive computing examination administered by the school district in order to be eligible to teach computing in the elementary schools (Aiken, 1988).

The model was designed to provide in-depth training in computing within two years while teachers continue their normal teaching duties. The program curriculum consists of five courses which have been designed to achieve all three project goals. The principal objective of the first and last courses is to prepare teachers to become computer resource people, while the middle three courses teach programming with a focus on how programming can be taught effectively to young children. Moreover, an underlying principle is that project participants would have a broader perspective about computer programming once they had seen what applications programs can do and how they function in the first project course.

The first course covers elementary computer concepts and introduces applications programs and their use in the classroom. The next two courses teach Logo programming at the beginning and advanced levels with an emphasis on modular and top-down programming concepts and on how problem solving using Logo can generalize to other types of problem solving tasks. The fourth course teaches problem solving with structured BASIC and prominantly features problem solving strategies, top-down design and structured programming concepts. The final course addresses educational software evaluation and effective uses of computers for teaching.

Participants are divided into two groups of equal size. Both groups follow the same course sequence with the second group starting a semester later than the first group. All courses are conducted in laboratory settings. Programming courses are taught in a laboratory of Apple IIe microcomputers. Since there are fewer computers than teachers, most teachers work in pairs during lab sessions. Elementary and middle school teachers from the District serve as project assistants and course teachers. Incentives include 15 free graduate credits (three graduate credits per course) and a $500.00 stipend for each participant who completes all five courses. In addition, participants are provided with free textbooks and supplies and as much computer time as needed and available in the Apple lab.

2.2 Preliminary Assessment.

Two groups of 26 participants were selected for this project from an applicant pool of 457. Forty-seven of the original 52 participants, or 90 percent, are expected to complete the project. On this criterion, the project has been more successful than believed possible.

There is also an early indication that the project will successfully motivate participants to seek positions as computing teachers at the elementary and middle school levels. Midway through the project, an official of the School District invited participants to apply for positions as computing teachers on a temporary basis. Approximately one third of the participants expressed interest and, within two months, several had been moved into such positions.

Other indications of the effectiveness of the current project are partial or anecdotal based on information available when the first group had completed all three programming courses and the second group had completed the two Logo courses. Some of the anecdotal information suggests that, although the project is achieving its goals generally, design modifications may be warranted.

Participants have performed well on programming projects, creating programs that are often quite innovative and enjoyable. Performance on midterm and final examinations also reveals that participants have a solid knowledge of programming concepts and of specific programming language features and a good understanding of how to apply this knowledge to solve simple problems.

In addition, participants appear to have mastered programming concepts and practices more quickly and completely than one would expect in a typical programming course. One of the BASIC programming instructors remarked that he had fewer problems teaching structured programming with BASIC to the first group than he usually has. This can perhaps be attributed both to the fact that they had been given a broad perspective and a top down view of computer programs during the first course and that programming was introduced using Logo which encourages a modular approach to program design.

First hand observations and informal discussions with participants and instructors during the programming courses confirm the anticipated benefits of the curriculum design. Preoccupation with programming language details does not appear to be as intense nor as extensive as had been the case in a previous teacher retraining project in which participants started to program in BASIC without any broad perspective about programming nor any real understanding of how programs function (Aiken & Snelbecker, 1990). On the other hand, some troubling behavior has been noted. It has been observed that participants are so highly motivated to succeed in the program and have such high expectations for their own performance that they are uncomfortable with less than perfect performance. This could be anticipated since participants were selected on the basis of high motivation to succeed. However, beginning and advanced Logo teachers have observed that these attitudes have been responsible for a significant degree of tension in the class. It has also been observed that participants inevitably become bogged down in programming language details, becoming frustrated and losing sight of the broader educational context in which these programming activities can provide stimulating learning experiences for children.

A somewhat more troublesome consequence of the participants' high self-expectations is the difficulty many have accepting and handling errors made while exercising new language constructs. As mentioned earlier, programming has been taught in a laboratory of Apple IIe computers with two teachers to a machine, encouraging continuous discussion and collaboration. As programming language features are introduced, participants are given simple exercises to perform which illustrate how these features work. In addition, participants are made familiar with general error recovery procedures and thus know that no error can be fatal.

During these lab sessions, discussion among participants often focuses in a nonconstructive way on errors, some participants becoming immobilized by simple mistakes. Many participants are unwilling to explore and play with new language features even when they are prompted with ideas for specific exploratory strategies. Perhaps participants have decided that the effort required to expore Logo features is not worth the trouble since such exploration seems irrelevant to completing assigned tasks or passing examinations. Another possibility is that they avoid exploring because they want to avoid making mistakes. Whether the structured focus of the course or the participants' high self-expectations best accounts for the observed error avoidance behavior, participants evidently have determined that exploration is not essential to learning to program with Logo.

In summary, objective indicators such as retention rates, and exam and project grades show that the model retraining program is an effective way to teach programming to experienced elementary and middle school teachers. However, anecdotal information suggests that the model may need to address more explicitly how teachers can learn to use programming knowledge to accomplish educational objectives in the classroom. This requires, minimally, a willingness to explore facets of programming languages without fear of making errors.

3. Teaching Teachers how to Teach Programming Effectively

One of the goals of the model retraining project for elementary and middle school teachers is to examine teaching strategies that maximize the general cognitive benefits of programming, such as development of problem solving and critical thinking skills that can be applied in both programming and nonprogramming domains. Objective measures of success used for this project do not directly measure the degree to which this goal has been achieved. Moreover, there is evidence that attitudinal factors may be interfering with its attainment. It is important to ensure that this educational goal can be met within the framework of the model program, and to develop ways of determining whether it has been achieved in practice.

3.1 The Problem.

As already noted, teachers selected to participate in the project are high achievers with exacting expectations for their own performance. While learning computer programming, a new and demanding subject for most of them, they have concentrated on mastery of the mechanics of the programming language, and many have become distracted by language detail, reluctant to explore new language features, and often appear tense. As is generally the case with introductory programming courses in universities, error avoidance behavior has been noted most commonly among participants with little or no previous computing experience, but tension and preoccupation with programming language mechanics are more prevalent.

These problematic attitudes and behaviors probably can be explained in terms of a combination of high self-expectation, lack of previous programming experience, and time constraints imposed by a full time teaching schedule. The problem is that these conditions cannot be changed under the constraints of the current retraining model.

Thus it is important to find other ways to help participants view errors as learning opportunities rather than as unpleasant obstacles and to minimize preoccupation with the mechanics of programming at the expense of general pedagogical objectives. Error avoidance behavior slows the process of learning how to program, but is even more worrisome in the case of future elementary school computing teachers for whom innovative, hence error-prone, approaches to programming will be an important educational tool. The primary objective is to prepare teachers to teach programming to children. Achieving this objective will require a good deal more than teaching the teachers how to program.

Ultimately, elimination of attitudes and behaviors that appear to interfere with learning may be insufficient. What is needed is some way to guarantee that the pedagogical goals of the project have been achieved. The problem is this: How can the difficulties of learning programming (often for the first time) be managed so as to permit the teaching experience of participants to be focused on the task of creating effective programming environments for children? Recent research on effective ways to teach and learn programming provide insight into how the project courses might be modified to meet this project objective.

3.2 Effective Programming Instruction Methods.

Through the 1970's, Seymour Papert and others claimed that children could experience general cognitive benefits from learning how to program (Papert, 1980). Papert advocated having children learn to program by freely exploring an interactive computer environment, a "microworld", whose simple objects and rules could be augmented indefinitely through programming. This "discovery" approach to learning to program contrasts with the standard approach where the instructor leads students through a structured syllabus. Papert believed that such an environment would enable programming expertise as effortlessly as living in France, say, enables fluency in French. Logo was designed to provide a discovery environment for learning programming.

By the mid 1980's, no conclusive evidence had been found to support claims that discovery learning with Logo promotes cognitive development any more effectively than do alternative instructional approaches, with or without programming (Pea & Kurland, 1984; Pea, Kurland & Hawkins, 1986). These results seemed to imply that little benefit would result from the sizeable investment in resources required to support teaching programming in elementary and middle schools. Recent research, however, indicates that more structured Logo environments, and, in particular, a mediated discovery environment, can facilitate childrens' mastery of programming and promote transfer of some problem solving skills learned in programming to other domains. In addition, several researchers have begun to describe characteristics of Logo programming environments that are likely to have specific cognitive benefits and to verify that these benefits do occur (Burns & Hagerman, 1989; Swan, 1989).

Recent studies of the cognitive benefits of programming with Logo compare unstructured (pure discovery), structured, and mediated learning environments. Each of these instructional approaches is laboratory based and includes initial presentation of new Logo content. Following content presentations, students in a pure discovery environment are free to explore Logo on their own, whereas students in a structured environment work on tasks assigned by the teacher. In the mediated discovery approach, instructors provide focused intervention and may assign or suggest some, but not all, tasks. Intervention is given to suggest problem solving strategies and to encourage development of metacognitive skills, helping students become aware of their own problem solving behavior, of its consequences, of alternative ways to achieve the same objectives, and of how problem solving strategies used in Logo programming may be applied to nonprogramming tasks.

Researchers are finding that general cognitive benefits can be achieved through programming only when children know programming well (Littlefield, Delclos, Lever, Clayton, Bransford & Franks, 1988; Pea & Kurland, 1984) and that structured methods seem to be superior to pure discovery methods for teaching programming well (Carver, 1988; Lehrer, Guckenberg & Sancilio, 1988; Littlefield et al., 1988; Swan, 1989). In addition, researchers are finding that mediated discovery methods, particularly those that include consideration of nonprogramming tasks, seem to achieve transfer of cognitive skills to nonprogramming domains more effectively than do either structured approaches, the pure discovery approach, or mediated approaches that are not computer-based (Lehrer et al., 1988; Littlefield et al., 1988; Swan, 1989).

A number of experimental studies have discovered measurable transfer of problem solving skills learned in programming with Logo to domains such as geometry, simple mapping problems, and error detection during a communication task (Au, Horton & Ryba, 1987; Carver, 1988; Lehrer et al., 1988; Littlefield et al. 1988). They also have demonstrated improved performance on measures of general problem solving skills, such as problem decomposition, error identification, and analogy (Keller, 1990; Swan, 1989). In addition, mediated Logo environments have been found to improve reflectivity (metacognition) and to foster divergent thinking (cognitive style) (Clements & Merriman, 1988).

Logo learning environments that are effective in developing general problem solving and metacognitive skills share the following characteristics:

(1) Sufficient time and explicit instruction is provided to ensure a mastery of Logo.
(2) Children are immersed in the problem solving process rather than the mechanics of programming.
(3) Instructors use focused intervention to encourage children to talk about problem solving processes in their own terms. These interventions use opportunistic questioning to prompt students to reflect on their problem solving processes (Au et al., 1987; Clements & Merriman, 1988).
(4) Explicit references are made to other domains such as ordinary planning activities or mathematics. Explicit references to familiar problems can reinforce mastery of problem solving skills in Logo and facilitate generalization to less familiar domains (Pea, 1989).
(5) Social interaction between teacher and child and between children is encouraged. Successful environments either encourage collaborative problem solving or schedule frequent discussions where children share accounts of the processes used to solve problems. Collaboration fosters conflict resolution and the discovery of better solutions. In addition, faster or more experienced children explicitly model reflective activity for other children. Such exchanges are collaborative rather than competitive (Clements & Merriman, 1988; Keller, 1990).

Transfer, which can be described in terms of generalizing from task specific situations to other knowledge domains, is fundamental to learning. Generalizable thinking skills have been successfully taught in mathematics (problem solving heuristics), in writing (planning and goal monitoring), and in reading comprehension (Pea, 1990). Learning environments that accomplish transfer in these domains share salient aspects of mediated discovery environments for Logo programming, namely, situated learning (knowledge and skills are taught in functional contexts), provision of multiple examples to help bridge to other domains, and development of metacognitive skills by modeling them and by encouraging students to explicitly emulate the model (Pea, 1990).

3.3 Reconsidering the Design of Programming Courses for Retraining Teachers.

Under the current teacher retraining model, programming is taught in laboratory courses that meet for three hours once a week. Language constructs and programming concepts are introduced in presentations which are interleafed with lab sessions where participants work on problems and projects assigned by the instructor. Thus programming is taught in a structured Logo environment as defined in the experimental literature. As noted in the literature, such environments seem to be more successful than unstructured environments at achieving programming mastery. The high standard of performance on examinations and programming assignments in project courses is consistent with these findings. On the other hand, mediated discovery environments appear to be as successful as structured environments at achieving programming mastery, yet more successful at achieving transfer.

These considerations suggest two alternative approaches to modifying the design of the project courses. Both approaches would place greater emphasis on instructional methods that foster development of general cognitive skills.

One approach would reduce the amount of time devoted to teaching programming languages by reducing the number of programming courses. Thus the program might be amended to teach one course each in Logo and BASIC or to eliminate the BASIC course. The remaining programming courses could then be followed by a course that explicitly treats cognitive issues in learning and explores the kinds of programming environments that tend to foster development of desired cognitive skills. Among other things, such a course could consider dialogue management, general problem solving strategies, and the embedded characteristics of programming environments that facilitate cognitive development (Burns & Hagerman, 1989) and enable transfer to related domains (Swan, 1989).

This approach would have the advantage of getting the details of the language out of the way so that participants could concentrate on pedagogical strategies in an instructional methods course. The disadvantage of this approach is that participants would have less programming instruction and so may fail to master programming sufficiently to feel comfortable teaching it.

A second approach to redesigning the programming courses would be to modify the existing courses so that they focus more explicitly on issues relating to how programming should be taught. This approach would not reduce the amount of programming instruction and so would not risk reducing the programming competency and confidence of participants. More significantly, it would allow teachers to experience the methods they would be learning, and later using to teach children. Since one tends to follow the behavioral models by which one learns, this approach is more likely to ensure that the new methods will be used.

Although problem solving strategies relevant to programming are emphasized in the courses as currently taught, participants still become preoccupied by programming detail at the expense of goals, strategies, plans, and overviews. A mediated discovery approach might help to counteract these inclinations. In such a mediated environment, teachers would be encouraged to monitor their own problem solving behavior and to monitor and guide the problem solving efforts of others (Clements & Merriman, 1988; Perkins, Schwartz & Simmons, 1988). They would also be encouraged to abstract problem solving strategies from the programming task (Littlefield et al., 1988).

The primary drawback of this approach is the effort required to teach it. It should be noted, however, that the equipment shortage which led to pairing participants on computers had unanticipated beneficial consequences. Participants often were able to help one another which allowed the instructor more time to engage in interaction with single participants or small groups. If pairing is done carefully to maximize collaborative interaction and to ensure that faster, or more experienced, participants are paired with slower, or less experienced, ones, then the instructor can assume more the role of a facilitator. One Logo instructor decided to change pairings half way through the semester. Although participant pairings had not been assigned they became set after the first few sessions and the instructor sensed that a few of these were dysfunctional because they reinforced non-constructive behaviors and attitudes. He had participants move anywhere in the lab so long as they had new partners. This change energized the class sessions noticeably.

4. Anticipated Benefits.

There are two reasons for modifying the program to include more explicit treatment of instructional methods and less treatment of the mechanics of programming. The first is that this better accomplishes one of the goals of retraining programs. The goals of any teacher training course in technology include both mastery of technology and of strategies for teaching the technology effectively. The revised curriculum would shift the focus from learning and learning how to teach programming languages to learning and learning how to teach general problem solving skills within a programming environment.

The second reason for focusing more on mediated learning strategies is to modify some of the attitudes noted among participants in the language courses as currently designed. The general idea is that by shifting the focus from mastery of programming to mastery of methods of instruction, the discomfort often experienced by novice programmers when faced with language detail can be managed.

This might be accomplished more effectively by integrating focused intervention techniques and peer collaboration into the programming courses and structuring these around problem solving strategies rather than programming language features. It seems likely that, having these courses focus on applying pedagogical skills which experienced teachers already have will alleviate the learning difficulties that have been observed, particularly the observed reluctance in many cases to explore the Logo environment. It also seems likely that instructional strategies that have been found effective in developing programming mastery for children can have similar benefits for teachers in a retraining program.

It should be emphasized that project courses so far have satisfied the objective measures of success for the model program and, in some cases, have far surpassed expectations. It is suggested that changes in the structure of the programming courses may contribute further to the model program's general effectiveness.

Acknowledgements

The work reported here has been funded under National Science Foundation Project Grant TPE-8850531.

References

Aiken, R., & Snelbecker, G.E. (1988). *A Model Program for Retraining Elementaryand Middle School Teachers to Use Computers Effectively in the Classroom.* Proposal for project funded by the National Science Foundation. Philadelphia: authors.

Aiken, R., & Snelbecker, G.E. (1990). Hindsight: reflections on retrainingsecondary school teachers to teach computer science. *Forthcoming in Journal of Research on Computing in Education.*

Au, K.W., Horton, J., & Ryba, K. (November 1987). Logo, teacher intervention,and the development of thinking skills. *The Computing Teacher,* 12-16.

Burke, W. (1990). Information technology in the 5 to 16 curriculum -a crosscurricular theme. In McDougall, A., & Dowling, C. (Eds.), *Computers in Education: Proceedings of the IFIP TC 3 Fifth World Conference on Computers in Education-WCCE 90,* Sydney, Australia, July 9-13, 1990 (pp. 31-36). Amsterdam: North-Holland.

Burns, B., & Hagerman, A. (1989). Computer experience, self-concept and problemsolving: The effects of LOGO on children's ideas of themselves as learners. *Journal of Educational Computing Research, 5* (2), 199-212.

Carver, S.M. (1988). Learning and Transfer of Debugging Skills: Applying TaskAnalysis to Curriculum Design and Assessment. In Mayer, R.E., (Ed.), *Training and Learning Computer Programming: Multiple Research Perspectives* (pp. 259-297). New Jersey: Lawrence Erlbaum Associates.

Clements, D.H., & Merriman, S. (1988). Componential developments in LOGOprogramming environments. In Mayer, R.E., (Ed.), *Training and Learning Computer Programming: Multiple Research Perspectives* (pp. 13-54). New Jersey: Lawrence Erlbaum Associates.

Goodspeed, J. (1989). The NEA Publishes two powerful reports on educationaltechnology. *Electronic Learning, 9* (1), 21.

Haney, M.R. (1990). Infusing technology into the high school curriculum -a fiveyear report from the Blair Magnet Program. In McDougall, A., & Dowling, C. (Eds.), *Computers in Education: Proceedings of the IFIP TC 3 Fifth World Conference on Computers in Education - WCCE 90,* Sydney, Australia, July 9-13, 1990 (pp. 109-113). Amsterdam: North-Holland.

Keller, J.K. (1990). Characteristics of Logo instruction promoting transfer oflearning: A research review. *Journal of Research on Computing in Education, 23* (1), 55-71.

Lehrer, R., Guckenberg, T., & Sancilio, L. (1988). Influences of LOGO onchildren's intellectual development. In Mayer, R. E., (Ed.), *Training and Learning Computer Programming: Multiple Research Perspectives* (75-110). New Jersey: Lawrence Erlbaum Associates.

Littlefield, J., Delclos, V.R., Lever, S., Clayton, K.N., Bransford, J.D., & Franks, J.J. (1988). Learning LOGO: method of teaching, transfer of general skills, and attitudes toward school and computers. In Mayer, R.E., (Ed.), *Training and Learning Computer Programming: Multiple Research Perspectives* (pp. 111-135). New Jersey: Lawrence Erlbaum Associates.

Papert, S. (1980). *Mindstorms: Children, Computers, and Powerful Ideas.* NewYork: Basic Books.

Pea, R.D., & Kurland, D.M. (1984). On the cognitive effects of learning computerprogramming. *New Ideas in Psychology, 2* (2), 137-168.

Pea, R.D., Kurland, D.M., & Hawkins, J. (1986). LOGO and the development ofthinking skills. In Pea, R.D., & Sheingold, K., (Eds), *Mirrors of Minds* (pp. 178-197). Norwood, N.J.: Ablex Publishing Corp.

Pea, R.D. (April 1989). Socializing the knowledge transfer problem. Institutefor Research on Learning, Report No. IRL89-0009.

Perkins, D.N., Schwartz, S, & Simmons, R. (1988). Instructional strategies forthe problems of novice programmers. In Mayer, R.E., (Ed.), *Training and Learning Computer Programming: Multiple Research Perspectives* (pp. 153-178). New Jersey: Lawrence Erlbaum Associates.

Swan, K. (1989). Logo programming and the teaching and learning of problemsolving. *Journal of Artificial Intelligence in Education, 1* (1), 73-92.

Special Presentation
Technology! Calculus! and Individual Differences!

Brent Wilson, Michel Pitre
Media Integration Program, Carleton Roman Catholic School Board
Nepean, Ontario

Technology! Calculus! and individual Differences! is an exciting classroom. It involves the integration of computers, tablets, videos, text, and software into every calculus course in all our high schools. It has received worldwide recognition by educators and school boards.

Developed by the Carleton Roman Catholic School Board, this exciting new approach to teaching and learning calculus includes sets of three software packages per unit, for use in conjunction with traditional and other teaching approaches. All software components generate questions randomly in such a way as to allow literally billions of variables, while using virtually no disk space! Tutorial sections are interspersed with detailed illustrations, animations, examples, highly visual interactive investigations and simulations.

These classrooms are designed to enhance individualized learning through the integration of high technology and multimedia products into the curriculum. It places curriculum and high technology products into the hands of the regular classroom teacher, thereby enabling the teacher to meet curriculum objectives, different learning styles, and rates of learning within the classroom environment . Student record keeping and tracking are facilitated through the software components.

As part of an overall school board initiative, these classrooms demonstrate the power and benefits of technology in the future of education.

Special Presentation—ISTE

Accreditation Standards for Instructional Computing and Technology Teacher Preparation Programs

This session presents the standards and guidelines developed by the International Society for Technology in Education (ISTE) Accreditation committee for evaluation of programs in instructional computing and technology. These guidelines address the standards and criteria for unit accreditation and folio approval that will be submitted for review by the National Council for Accreditation of Teacher Education (NCATE) in September 1991.

Research providing the basis for development of the standards includes competencies for preparation of all teachers, administrators, and candidates in programs at the Bachelor, Master, and doctoral levels. results of the study and the process teacher education programs follow when seeking accreditation for their programs in instructional computing and technology are discussed and examples of activities meeting the standards given. The panel of presenters including members from the NCATE/ISTE Liaison, and ISTE Accreditation Committee are available to answer questions regarding the standards and guidelines for unit and program accreditations.

Panel Members:

Don Knezek
Accreditation Committee
Education Service Center—Region 20
San Antonio, TX

C. Dianne Martin
ISTE/NCATE Liason
George Washington University
Washington, DC

Sally Sloan
Accreditation Committee
Winona State College
Winona, Minnesota

Harriet Taylor
Accreditation Committee
Louisiana State University
Baton Rouge, Louisiana

Lajeane G. Thomas
ISTE Representative to NCATE Board
Louisiana Tech University
Ruston, Louisiana

James Wiebe
NCATE Board of Examiners
California State University, L.A.

Special Presentation

The Texas Center for Educational Technology: Current and Future Solutions

Kenneth E. Brumbaugh
James L. Poirot
Texas Center for Educational Technology

The Texas Center for Educational Technology was created in June, 1990, by the Texas Education Agency at the University of North Texas, in collaboration with the University of Texas at Austin. The major goals of the Center are to:

- create a statewide exchange program for scholars, teachers, researchers, and industry personnel;
- expand summer in-service teacher training programs for using technology in the classroom;
- create model classrooms to demonstrate the latest in technology-based teaching and learning;
- have the nation's strongest educational computing and technology-based research effort.

The Center's eleven research and development laboratories cover the following areas:

1) Assessment of Student Learning and Cognition
2) Student Learning and Special Populations
3) Teacher Productivity and Training
4) Teacher Retention and Support Networks
5) Curriculum and Instruction
6) Instructional Design and Evaluation
7) Educational Telecommunications and Informatics
8) Multimedia and Emerging Delivery Systems
9) Design of Computer-Based Instruction
10) Brain Mapping and Evoked Potential
11) Collaborative Software Development

TCET serves the Texas Education Agency, foundations, community colleges, coalitions, universities, professional organizations, businesses, and school districts statewide. It will research new technology applications for education, create prototypes of educational applications for technology developed for commercial or other purposes, and develop classroom applications for current technology.

Projects

Technology and Special Populations

The CLC Technology Project: Meeting Literacy Needs of Special Populations

Jane D. Steelman
Carolina Literacy Center
Louisburg, NC

Abstract

The Carolina Literacy Center, created under a grant received by the Department of Medical Allied Health Professions at the University of North Carolina at Chapel Hill, is the first facility of its kind at the state or national level to focus attention on the literacy needs of individuals with severe speech and physical impairments. (SSPI).

The center is committed to addressing the literacy needs of the more than 10,000 SSPI individuals in North Carolina. An estimated 90 percent of such individuals read at levels significantly below their nondisabled peers. The achievement of literacy opens doors to academic and vocational advancement for these individuals.

The project presentation will describe the technology project and its stage of development at the time of the presentation.

Center for Special Education Technology

Susan Elting
The Council for Exceptional Children
Reston, VA

Abstract

The Center for Special Education Technology at the Council for Exceptional Children is a national information center funded by the U.S. Department of Education, Office of Special Education Programs. Through its active participation and development of vital information products, the Center has become a national leader in the area of special education technology.

CENTER SERVICES

As a national resource with the goal of increased use of information about technology in the education of children with disabilities, the Center provides theme-related and general information services.

Theme-Related Services. A major component of the Center's services is its in-depth focus on identified themes in special education technology. theme activities address expressed information gaps by identifying existing knowledge and organizing that knowledge in various product formats. To date, five information themes have received Center attention.

The Center's efforts in assistive technology focus on products and services that provide access within the educational/learning environment. In the area of funding, information products emphasize awareness of funding sources and strategies for the acquisition of technology in educational settings. The technology training theme emphasizes the infusion of technology in the process, content, and products of preservice and inservice training. Technology integration focuses on factors that contribute to the successful integration of technology in educational systems. Technology for the moderately handicapped student explores the potential of technology-assisted instruction for this population.

General Information Services. To meet the continuing need for general awareness information about the use of technology and alternate resources in the field, the Center provides ongoing general information services.

Current products include:

State and topical Resource Inventories listing local, state, and national technology resources that provide information and services for professionals and consumers. A Tech Use Guide series summarizing issues and applications in technology use such as computer Access, Technology for Preschool Children, and Computers and Cooperative Learning. the Marketplace is a periodic publication highlighting vital marketing issues affecting special education technology. TECHLINE is the Center's electronic newsletter on special Net which provides current and topical information to special education professionals and education agencies.

SUMMARY

The Center's information products are targeted for service providers and product providers active in the use of technology in special education. These audiences in turn provide information and support to the technology consumers—educators, parents, and individuals with disabilities—who ultimately apply the technology in living and learning. The Center's products are in the public domain and are routinely made available to representatives of target groups and to individuals on request.

I Care; You Care: Together, *You* Can Do It!

Diana M. Willis
Suwannee District Schools, Live Oak, FL

Abstract

This project centered around youth in a group child care facility for dependent, neglected, and abused children. These youth are, on the average, two academic years behind their peers. This is one of the many reasons these youth are missing the incentive to learn and to excel. These "at-risk" youth are in danger of failing to complete their education with an adequate level of academic and interpersonal communication skills.

The project used computers, software written especially for "at-risk" youth, extensive counseling sessions, and a pre/post attitude survey.in an effort to foster through positive learning experiences the growth of both self-esteem and adequate levels of academic skills. It was hypothesized that by building self-esteem, attitudes toward school and behavior toward peers would change measurably.

The project was partially funded through a grant from the Suwannee Foundation For Excellence In Education, Inc. and through in-kind contributions of other business partners. The presentation will share information concerning the final status of this phase of the project.

Paper

The Nihongo Tutorial System for Learning Technical Japanese

Nelson K. Leung
Anthony A. Maciejewski
Purdue University

Abstract

The Nihongo Tutorial System (NTS) is an Intelligent Computer Aided Instruction (ICAI) system designed to assist western scientists and engineers in developing a reading competence of technical Japanese. The goal of the NTS is to maximize comprehension through context by providing individualized lessons that conform to the student's technical interests. It makes use of an augmented parse tree data structure to store syntactic, semantic, phonetic, and morphological information about an article, and incorporates a student model to represent the student's interests and current Japanese proficiency. In addition, the tutorial system supplies several electronic dictionaries for ease of cross-referencing. Through the NTS, the student is able to learn the general concepts of usage, meaning, and pronunciation of Japanese text as well as the style of technical Japanese writing.

This material is based upon work supported by the National Science Foundation under Grant Number INT-8818039. The Government has certain rights in this material.

Introduction

Although interest in learning Japanese is increasingly growing, many English-speaking people find it very difficult to learn the language due to its complexities. The situation is even worse for those western scientists or engineers who would like to read technical publications published in Japanese because of a number of factors. First, there are very few textbooks available to illustrate the style of technical Japanese writing. Second, textbooks usually do not cover specific technical disciplines in detail. In addition, the student cannot rely on using Japanese technical journals for self-study because only a small percentage of such publications are also available in English. Third, there are very few instructors who are fluent in both Japanese and English in addition to having expertise in a technical area. Finally, since most learners of technical Japanese are already engaged as professionals, they cannot spare a great deal of time and effort attending a structured class to study the language. To address these problems, a computer based tutorial system is an effective solution because the computerized tutor can automate most of the instructional tasks and provide relevant instruction to meet the diverse needs of technical Japanese students.

The Nihongo Tutorial System (NTS) described here is an Intelligent Computer Aided Instruction (ICAI) system developed to assist people in developing reading competence of technical Japanese. The goal of this tutorial system is to maximize comprehension through context, and to minimize rote memorization and dictionary use. The NTS is composed of three components: the Nihongo Tutor, the Parse Tree Editor and the Administrator. The Nihongo Tutor helps the student to learn the general concepts of usage, meaning, and pronunciation of Japanese text by providing technical Japanese articles for study. It also keeps a record of the student's current Japanese proficiency level in his personal database. The Parse Tree Editor, which is the authoring tool for this ICAI system, is used to prepare information about each article for the tutor by incorporating syntactic, semantic, phonetic, and morphological information into the parse tree of each sentence. The Administrator performs automatic article selection for the student by assigning an appropriate article that matches his current Japanese language proficiency and his technical areas of interest. This presentation focuses on a description of the Nihongo Tutor. A detailed description of the Administrator and the Parse Tree Editor can be found in the literature.[1]

The NTS was developed to work on the Macintosh line of computers due to their desirable human-machine interface and cost effectiveness. This also allows it to take advantage of the available Japanese Interface System, called *KanjiTalk*, to assist in the input, display, and editing of Japanese characters.

Overview of the Japanese Writing System

The Japanese writing system is composed of two distinct orthographies known as Kanji and Kana. The Kana are further divided into Katakana and Hiragana. Kanji are logographic characters, originally adopted from the Chinese language, for writing the vast majority of Japanese words such as nouns, verbs, and adjectives.[2] An example of a Kanji compound is 日本語 (nihongo), meaning the "Japanese language." Although there are more than 50,000 Kanji characters in existence, only about 2,000 of them are designated by the Japanese Government as official "Jōyō Kanji" for daily use.[3] These Jōyō Kanji account for approximately 90% of the Kanji found in standard texts.

Each Kanji has two kinds of readings: the "on" and "kun" readings. A single Kanji may have two or more readings, each indicating a different meaning for the character. For instance, the kun reading of 語 is かた(る) (kata(ru)), which means "talk, relate," whereas its on reading is ご (go), meaning "word" in Japanese compounds.[4] Each Kanji consists of a series of strokes written in specific directions. The stroke count, 14 for the character 語 as an example, is one of the traditional methods to categorize Kanji. Most Kanji characters can be divided into smaller recurring components. Over 90% of Kanji have two parts: the radical indicating the general area of meaning of the character, and the other representing the phonetic component. The radical of the character 語 is the component on the left-hand side, 言. A common way to search for a Kanji in a dictionary is to use its readings, stroke count, radical, or a combination of the above.[5]

Unlike the Kanji, the Kana are phonetic symbols developed to represent Japanese syllables. The meanings of the original characters are ignored. The Japanese devised the two separate syllabaries, Hiragana and Katakana, to simplify the process of phonetically writing Japanese text. They are two different scripts for writing the same sounds. For example, writing 日本語 phonetically in Hiragana results in にほんご. Katakana are mainly used to transliterate words borrowed from foreign languages whereas Hiragana are used to write native Japanese words such as inflectional endings of conceptual words like verbs, particles, auxiliary verbs, and also words that are primarily grammatical in function.

Learning to read Japanese is not an easy task if one considers all of the different aspects of the language such as the large number of characters, the large number of homonyms, the lack of word delimiters, the complexity of character images, and the complex lexical principles used to combine the three character sets. Many of these problems complicate the usage of Japanese in computers as well. However, the Japanese have devised a standard set of computerized Japanese characters known as the Japanese Industrial Standard (JIS) to allow for the display and use of Japanese in computers. The Japanese have also developed methods to transliterate Roman characters into Kana characters as well as for converting Kana characters into Kanji.[6]

Intelligent Computer Aided Instruction

Given that the Japanese language is so difficult for Westerners to study, the NTS has been developed to make use of computer resources to assist instructors in teaching technical Japanese. It is an intelligent CAI system that uses knowledge representation and deduction about the domain and each student to control most aspects of the pedagogical interactions with little assistance from an instructor.

A CAI system usually contains two parts: authoring system software and delivery system software. The authoring system software provides a system by which courseware can be created or edited. The delivery system, on the other hand, controls the interaction between the student and the tutorial system by providing subject material, recording the student's responses, and so on.

Three important components of an ICAI system are the representation of the subject material, student modelling, and the instructional strategy used. ICAI is usually targeted towards complex learning domains so that the instruction programs are flexible and generative (that is, they can be run repetitively by the same student although his learning competence is different each time). In NTS, instruction is individualized by using a student model to account for the student's knowledge and capabilities.[7] By storing information about each individual student (such as his current proficiency level, his areas of expertise and interest, etc.), NTS selects relevant subject material and relates his responses to his competence model. The instructional strategy used by NTS manages to organize tasks like topic selection or student evaluation into an intelligent process so that the most appropriate lesson will be assigned to a student.

Thus far, very few computerized instructional aids have been developed for learning Japanese. Those that have been described in the literature include the New Direct-Reading Japanese-English Character Dictionary[8] by Jack Halpern, KanjiCard[9] by Kazuko Nakajima and KanjiMaster[10] by Dezso Benedek. These systems are designed primarily to teach people to understand more about Kanji, providing extensive information about the meaning, reading, usage, and graphic features of individual Kanji characters. In contrast, NTS puts more emphasis on Japanese syntax, the contextual use of Kanji, the correlation between the Kanji and Kana, and showing the style of technical Japanese writing. Thus the philosophy of NTS is comparable to that of the Japanese CAI project at the University of Pittsburgh.[11]

The Nihongo Tutor

The Nihongo Tutor is the delivery system software of the NTS. Its underlying task is to enhance the student's proficiency of technical Japanese by providing him Japanese articles to study. The student is expected to learn the general concepts of usage, meaning, and pronunciation of Japanese text as well as the style of technical writing through the tutor. The purpose of the tutor is to optimize comprehension through context, and to minimize dictionary lookup and mechanical translation. The article assigned is carefully selected according to the student's current Japanese competence, the level of difficulty of the article, and his technical area of interest. Presumably, if the student does not know the meaning or pronunciation of some portion of the text's contents, he will request semantic, syntactic, or phonetic information from the tutor. Consequently, the tutor uses a parse tree data structure to represent the subject material in order to provide fast and efficient interaction between the student and the tutor.

The tutor also furnishes several electronic on-line dictionaries for cross-referencing purposes: a character dictionary, a Kanji compound dictionary, a grammar dictionary and some specialty dictionaries. The tutor keeps a record of what the student learns from the current lesson in order to maintain and support student modelling. It stores all material that the student masters in addition to that with which he has difficulty. New information is stored into the student database whenever he finishes studying a lesson. This kind of database management helps the system to assign an appropriate article for him as the next lesson.

Representation of the Subject Material

For natural language processing, each sentence in an article can be represented by a parse tree based on the syntax of the language. The semantics of the sentence is also augmented into the parse tree as additional information stored in each node. Figure 1 shows how a Japanese sentence is represented in a parse tree. Note that each node in the parse tree may have an arbitrary number of child nodes.

A generalized tree is a suitable data structure to represent the augmented parse tree. In the NTS, this generalized tree is implemented as a doubly-linked list. The linked list can effectively connect all nodes together for fast searching and retrieval. Moving upward or downward along different levels of the parse tree is accomplished readily due to the double link between the nodes. This allows moving downward along a parse tree in order to show how a sentence or a phrase is broken down into its constituents, in addition to moving upward to depict how several clauses are combined together to form a sentence. The double link thus allows the student to investigate Japanese sentence structures more easily. This is especially important when learning Japanese since there is no character used to denote word or phrase boundaries. Thus, it is sometimes difficult to understand the Japanese sentence structure because the first step of breaking a sentence down into phrases is not trivial.

Original Japanese sentence:

速度というのわ速さと向きで表わされるものである。

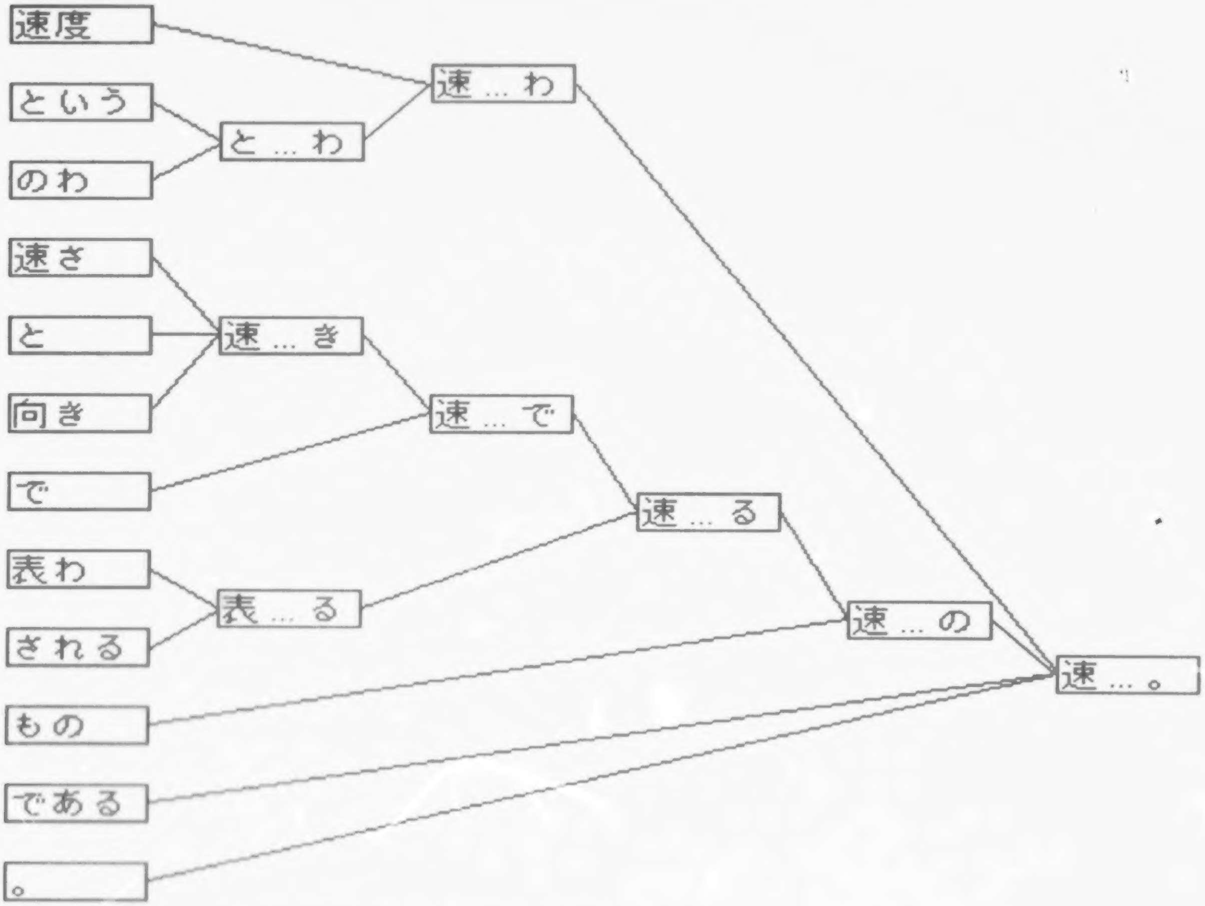


Figure 1. A Parse Tree Representation of a Japanese Sentence

With the help of this parse tree data structure, translation information is retrieved by the scheme shown in figure 2. Whenever the student has difficulty in understanding some of the text in the article, he highlights this text to notify the tutor. To get the translation information, the Nihongo Tutor locates the highlighted text in a sentence and returns the associated parse tree. It then finds a node in the lowest possible level of the parse tree enclosing all of the selected text. All the translation information based on this node is retrieved and then displayed.

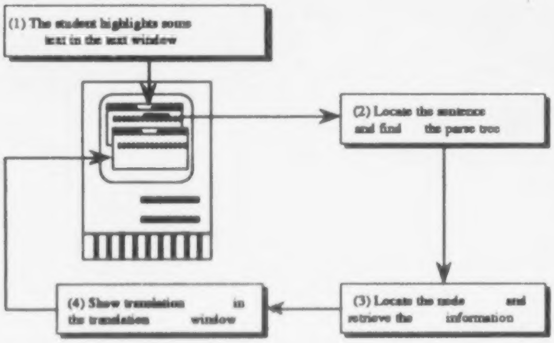

(1) The student highlights some text in the text window
(2) Locate the sentence and find the parse tree
(3) Locate the node and retrieve the information
(4) Show translation in the translation window

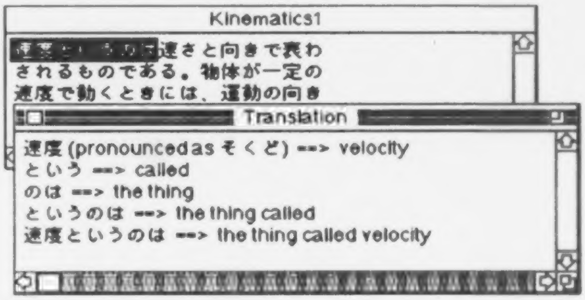
Figure 2. Interaction between the User and the Nihongo Tutor

The attributes of each node in the parse tree incorporate all semantic, phonetic, and morphological information, as well as any other supplemental data for instruction, in a compact form in order to remove any redundancy. In particular, they include the lexical boundary of the stored text, its best English-equivalent contextual meaning, Japanese readings or pronunciations, possible syntactic breakdown, and so on.

Syntactic Analysis

The tutor provides a utility, called syntactic analysis, in order to assist the student in learning about the grammatical rules of Japanese. This is accomplished by showing how a clause or a sentence is split into its basic constituents. With the help of this kind of breakdown, the student can understand more about the syntactic structure of a Japanese phrase, in addition to its overall meaning.

Figure 3 presents an example of syntactic analysis. The task is achieved by moving upward along the parse tree. Going to the parent of a node automatically includes all the sibling nodes. To show how several words or phrases are combined together to form a clause, the tutor uses a double-click facility together with the Macintosh mouse. If a double-click occurs when the mouse pointer is on some highlighted text, the selection range will be changed to that of the parent node. This happens because a node in the upper level of the parse tree is being selected. Similar double-clicks will move the selection upwards until the root node is reached. If syntactic analysis is requested, the translation information of all the descendant nodes will be displayed. This is designed to enhance the student's comprehension of word or clause boundaries in a Japanese sentence.



速度というのは速さと向きで表わ
されるものである。物体が一定の
速度で動くときには、運動の向き

速度 (pronounced as そくど) ==> velocity
という ==> called
のは ==> the thing
というのは ==> the thing called
速度というのは ==> the thing called velocity

Figure 3. Example of a Syntactic Analysis

Electronic Dictionaries

In Japanese, it is often the case that there will be more than one meaning associated with a Kanji. The exact meaning in an article depends on the context of its usage. The general contextual meanings of Kanji can be acquired by comprehending usage instead of by rote dictionary memorization. Ideally, the design of the tutor is such that use of the dictionaries will be minimized. However, if a Japanese word is used in several articles with various meanings, then an electronic on-line dictionary is valuable as it provides the student a correlation between the different usages of the word.

The set of electronic dictionaries in the NTS includes a character dictionary, a Kanji compound dictionary, and even several specialty dictionaries, built up with specific technical terms from various scientific disciplines. In the current version of the implementation, the character dictionary contains 1,945 entries, including the basic Jōyō Kanji, along with their most important readings and definitions. The compound dictionary currently incorporates 1,000 common Kanji compounds which are the most frequently occurring compounds appearing in Japanese newspapers and books.[12] The specialty dictionary is composed of a group of special dictionaries, each devoted to a specific technical discipline. For instance, the specialty dictionary in the tutor now contains a communications dictionary encompassing 8,889 entries of communication terms. In addition to the character, compound, and specialty dictionaries, there is also a grammar dictionary with 346 grammatical entries.[13]

The goals of the electronic dictionary can be summarized as follows:

(1) To provide the student with an understanding of the meanings, readings and characteristics (stroke counts, radical

number, Nelson's dictionary reference number) of Kanji and Kanji compounds.

(2) To enable the student to look up Kanji characters and compounds with high speed and different search schemes.

(3) To facilitate a cross-reference between the translation in an article and the definitions in the dictionary.

Each entry in the character dictionary includes a Nelson's dictionary reference number, a radical number, a stroke count, common meanings of the character and their associated readings. The Nelson's dictionary reference number is an index to *The Modern Reader's Japanese-English Character Dictionary* by Andrew Nelson, one of the earliest Japanese-to-English dictionaries which many students use for reference. Since a Kanji has many properties, the student is allowed to use different searching criteria to cross-reference Kanji characters with similar characteristics. For example, when none of the readings of a Kanji are known and determining its radical number seems too troublesome, the character can be located by means of its stroke count.

For each dictionary search, all meanings and readings related to the search key are displayed. The possible access keys are by Kanji, readings (on and kun), Nelson's dictionary reference number, radical number, stroke counts, and English translation. The English translation key offers a method for searching Kanji characters based on their English definitions. The Nelson's dictionary reference number, the stroke counts, and the radical number of a Kanji are all unique; however, this is not the case for its on and kun readings.

The Kanji compound dictionary consists of common Kanji compounds, including their meanings and the associated pronunciation. The search key is either the entire Kanji compound or one of its individual Kanji. If the latter is provided for searching, all relevant compounds are retrieved so that a cross-reference can be obtained. The grammar dictionary has an English translation, a description of the grammatical structure and one or more examples for each entry. Each example contains a Japanese sentence illustrating the usage of the entry, followed by a sentence showing the English translation for that Japanese sentence.

Student Model

To monitor the student's progress in Japanese proficiency, the tutor records important performance information to aid the Administrator in deciding the student's competence. Such information includes material that the student knows well, as well as those items not currently mastered. This information is in terms of Japanese words, phrases, and even general grammatical structures. Each time the student highlights some Japanese text and requests translation information, the tutor updates his personal database so that he will be presented with articles appropriate for his level of competence in the next lesson.

One of the difficulties in building the student model of language proficiency is evaluating why a student has asked for a particular piece of information. One direct way to evaluate the student's competence is to query the student to see if he has difficulty with a phrase whenever he requests translation information. However, this is too time-consuming and would interrupt the process of instruction in order to gather accurate data for assessing proficiency. Thus, the trade-off of not collecting such data to maintain the flow of study is tolerated. The evaluation scheme used in the tutor is to classify textual material in an article into two areas: well-known and unfamiliar. The first one keeps a record of all material that the student has mastered whereas the second one records those items that he has been exposed to yet still has difficulty comprehending. In order to classify material into either the well-known or unfamiliar area, the tutor makes the assumptions that the student requests translation information for some Japanese text if and only if he does not understand it. As a result, this material will fall into the unfamiliar area and any other material in the text will be classified as well-known. Clearly, these assumptions can be violated in practice; however, it is a reasonable approximation to the way that most students behave.

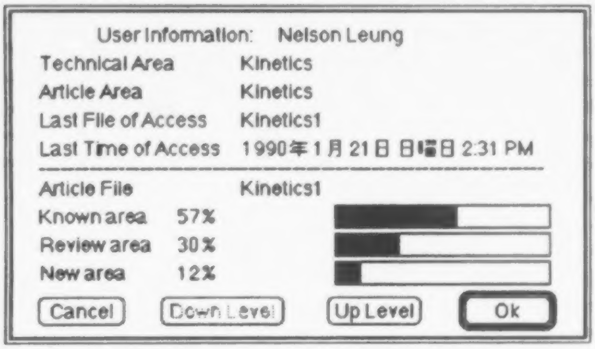
Since a Kanji may have various readings in different contexts, it is important that these readings are recorded to reflect what readings the student currently knows. Whenever he learns new readings of a word, they are added to update his database. Each entry in the unfamiliar area has a frequency counter which is a measure of the difficulty that the student is having with understanding this entry.

Student Evaluation

In practice, there is no clear-cut division between material that is known or unknown so that these databases should not be independent of each other. For instance, if a student learns some new Japanese text, it is initially put under his unfamiliar area. After a few lessons which include the same material, the tutor should transfer it from the unfamiliar area to the well-known area if the student does not once again ask for the translation. On the other hand, any material in the well-known area should be moved to the unfamiliar area if the student requests information on this material. Hence the evaluation scheme accurately reflects the student's change in competence by handling both the case where the student learns more about Japanese as well as the case where he forgets some text that he has studied before.

The evaluation scheme in the NTS adds, updates, or transfers entries to the unfamiliar area whenever the student asks for a translation. Any text for which the student does not request information is added or transferred to the well-known area after he finishes reading the lesson. There is no guarantee that this material is necessarily well-known by the student; however, putting it under this area does not give rise to a problem. If this material really should fall into the unknown area, the scheme will transfer it once the student requests translation information in future lessons. Thus, this scheme resolves conflicts so that the same Japanese text in different lessons would not be put into different areas. The records which hold the information for the known and unfamiliar area are loaded into the system in the form of a hash table data structure in order to support rapid access to this large amount of data.

The selection of an appropriate article for the student is performed by the Administrator software which makes use of the student model along with information about the student's technical interests. In particular, it compares the student's personal database with the lexemes of the article, (i.e., all leaf nodes in the parse trees of the sentences), in order to give a rough estimation of how much new material will be presented to him. An example of information used for article selection is displayed in figure 4. The instructional strategy is to assign an article that has a specific amount of new material, thus precluding unnecessary repetition of already mastered materials and maximizing comprehension through context.



User Information: Nelson Leung

Technical Area	Kinetics
Article Area	Kinetics
Last File of Access	Kinetics1
Last Time of Access	1990年1月21日 日曜日 2:31 PM
Article File	Kinetics1
Known area	57%
Review area	30%
New area	12%

Figure 4. A Comparison of a Student's Database with the Lexemes of an Article

Conclusions and Future Work

As previously stated, the main goal of the NTS is to assist scientists and engineers in developing reading competence in technical Japanese. The system is currently being tested at a selected number of universities. Future work includes further enhancement of the NTS such as the incorporation of more electronic dictionaries and more dictionary entries, automatic translation assistance of technical Japanese articles, and the preparation of a rich set of instructional texts.

Bibliography footnoted in text

1. Leung, K. W. (1990, May). *The Nihongo tutorial system for learning technical Japanese.* Masters Thesis, School of Electrical Engineering, Purdue University, West Lafayette, IN.
2. Yasuo, Y. (1973). *Japanese for today.* Japan: Gakken Company, Ltd.
3. Unger, J. M. (1987). *The fifth generation fallacy.* New York: Oxford University Press.
4. Hadamitzky, W. & Spahn, M. (1981). *Kanji and Kana.* Vermont: Charles E. Tuttle Company, Inc.
5. Nelson, A. (1962). *The modern reader's Japanese-English character dictionary.* Vermont: Charles E. Tuttle Company, Inc.
6. Payne, P. & Utsuru, Y. (1989). *MacKanji 6.0 user's manual.* Washington: Linguist's Software, Inc.
7. Rickel, J. (1989, January/February). Intelligent computer aided instruction: a survey organized around system components. *IEEE Trans. on Systems, Man, and Cybernetics,* 19(1), 40-57.
8. Halpern, J. (1985, December). New direct-reading Japanese-English character dictionary. *CALICO Journal,* 3-6.
9. Nakajima, K. (1988, December). On developing HyperCard stacks for the study of Chinese characters: KanjiCard. *CALICO Journal,* 75-87.
10. Benedek, D. (1990, April). Computers for reading and writing Japanese. *42nd annual meeting of the Association for Asian Studies,* Chicago, IL.
11. Nara, H. (1990, April). Issues in developing a Japanese CAI. *42nd annual meeting of the Association for Asian Studies,* Chicago, IL.
12. Kikuoka, T. (1970). *Japanese newspaper compounds.* Vermont: Charles E. Tuttle Company, Inc.
13. Makino, S. & Tsutsui, M. (1986). *A dictionary of basic Japanese grammar.* Japan: The Japan Times, Ltd.

Production note:

As you'll note, some of the text in this paper appears different from the usual text styles used throughout the *Proceedings*. This is because parts of this paper came to us in a special Japanese operating system called *KanjiTalk 6.0*, used with *MacWrite*. The result is that the sections containing Japanese characters are reproduced as bitmapped graphic; hence, the different appearance. We thought this distinction interesting, and left it as you see it here for your information.

Ian Byington
ISTE production assistant

Paper

"What did he say?": Speech Output in Preschool Software

Erik Strommen
Children's Television Workshop
New York

Abstract

Advances in technology have made it possible to include speech in electronic learning materials for small children. Little is known, however, about how young children respond to speech as a component of technology. Based on the results of a series of studies of Sesame Street interactive learning games tested with over 150 children aged 3 to 5, guidelines for evaluating speech as a feature of computer software are presented. Preschoolers responded best to speech segments that were short, contained concrete language, and were not stereotyped or repetitive. Information presented solely via speech was much less effective than when speech segments were combined with supplemental, task-related visual information that reinforced their content. Guidelines derived from these results can be used for judging the quality and effectiveness of speech included in interactive technologies such as computer software and videodiscs.

Recent and rapid changes in technology have made speech a possible feature of interactive electronic learning materials. To date, "talking machines" have most commonly been studied as aids in reading and writing tasks (c.f. Parham, 1988; Reitsma, 1988; Weiner, 1991). However, the potential uses of speech in applications for small children, not as an adjunct to task content but as an effective form of interface, have been given very little study. Because young children cannot benefit from onscreen text messages, spoken requests and feedback can potentially empower young children's learning in complex tasks that currently require verbal guidance and explanation by an adult. Such uses of speech as an interface, however, must be carefully considered. There is abundant evidence that young children have difficulty decoding spoken language in everyday contexts (Beal and Belarad, 1990; Hargrove and Panagos, 1982; Markham, 1979), so issues of how best to integrate speech into interactive multimedia technologies must be carefully considered.

The selection of the nature and content of speech in computer systems is especially important because unlike human speakers, who can tailor their speech to their listeners in an endless variety of ways, the affordable computer of the near future will typically only be able to repeat a limited variety of stock phrases. How long these phrases should be, how they should be worded, and where they should be placed as part of the interactive task, are issues of vital importance if speech is to be used as an effective, engaging component of educational materials for pre-literate children.

The present paper summarizes the results of a series of seven formative studies on computer learning materials for young children that employ speech as an integral component. The goal of this summary is to describe how children reacted to speech as a component of the computer activities, and to make a first attempt at identifying some guidelines for the optimal length, content, and placement of speech in interactive learning materials. These guidelines can serve designers of software using speech, but more importantly they can be used by teachers and others who need to evaluate new products for their suitability with their children.

Method

Participants

One-hundred and fifty children ranging in age from 34 to 62 months, drawn from the greater New York area, served as subjects. Equal numbers of boys and girls were tested.

Materials

Prototypes for seven educational software games for preschoolers, using Sesame Street characters and settings, were used in the present study. These games were designed to teach a variety of familiar preschool skills, such as rhyming, simple arithmetic, etc. Each was identified with a specific Sesame Street character who served as the game's "host" in an animated setting similar to familiar Sesame Street settings (Ernie and Bert's house, Big Bird's nest, etc.). The scripts for the games were written based on conventions from the Sesame Street television series, and were designed to cover the possible alternatives that crop up in interactive tasks: initial instructions, feedback messages, prompts, and end-of-game comments. The spoken portions of the games were recorded by the actual Sesame Street characters, and then incorporated into the games via digitized speech systems available on standard IBM DOS computers.

Six of the games were designed with the same broad format. To play the game, children placed a cursor on the muppet and pressed a button to start game play. The character then explained the game (the muppet animated during spoken segments), and play commenced. In four of the games, the spoken instructions contained crucial game information (words to find rhymes for, attributes to match with objects, verbal math problems, etc.); in two others, the spoken feedback from responses contained the crucial information. All six of these games also relied almost exclusively on spoken dialogue from the characters to convey incorrectness/correctness of responses, repeat key information, and prompt the child. The response structure was also the same for all six games: The child was required to move the cursor to objects and select them as the basic method of game play. If, for example, Ernie asked the child to "Find things that are blue," the child would move the cursor to an object and click on it; a correct response received a reward animation and Ernie saying "My airplane! It's blue! Good going!" and a randomly varied prompt encouraging the child to find another object that is blue. The child would go on to click on the next object, and so on.

The seventh game tested was slightly different, and involved the child "phoning" different muppets and listening to verbal segments reinforcing basic counting, addition, and initial-sounds concepts.

Procedure

The children, with an adult Researcher as "helper," all played multiple rounds of each game, with each child playing only one game each. The children's interactions with the games were videotaped, and responses to the games, especially to the verbal materials, were scored for the presence of confusion or hesitation, need to repeat spoken segments, requests for assistance from the adult, and affective reaction. The children's spontaneous comments were also recorded.

Results and Discussion

While the results vary somewhat depending on the particular game being described, six consistent findings relating to the use of speech recurred across all seven of the games tested.

1. It appears that instructions lasting 20 seconds or less, and feedback messages and prompts lasting less than ten seconds in length, are ideal. Children tended to ignore, or fail to attend to, important content in a long speech stream. While adults had no trouble picking up the important information, the children often missed it entirely, responded only to the last few words, or reacted only to very simple, concrete language such as "...press the button!". Such performances may reflect a bias by young children toward expecting that communicative messages are meant to prompt an action (Ackerman, 1981; Shatz, 1978).

2. Children tended to respond to the computer as if the character was actually speaking to them, particularly when action words or commands were spoken. If the character queried "Are you ready?" children invariably answered "Yes!" as if they could be heard. Such prompts proved to be very beneficial for cuing children that important verbal information was about to be presented, as did the prompt "Watch this!" for cuing the child to important visual information, and "move your cursor to..." for prompting game action.

3. The children responded to the inflection of the language as much as to the content of what was said. Exuberant language on correct response feedback, such as "That's right!" or "Hooray!" was highly reinforcing for children, and low-key or questioning affect ("My hat?") was effective for feedback to an incorrect response. It was also found that negative feedback, given exuberantly, was often misinterpreted as implying correctness even though the actual content of the speech belied the inflection.

4. Spoken prompts stating the names of objects as they are highlighted can facilitate game play. In several of our games, children were required to select objects based on their name (finding a number six, locating a rhyming object, etc.). Having the object's name spoken if the child stops the cursor on it for a few seconds proved to be an excellent way to support the performance of children who could not identify the objects in question.

5. Children are sensitive to stereotyped or repetitive speech. Children frequently asked "Why does he always say that?" or "I know already!" after a few rounds of game play. They often begin to talk over, or ignore, phrases they had heard repeatedly even if they did not understand the content of the message. Having speech-based systems employ a variety of phrases for the same situation (having four varied error messages, for example) would be beneficial.

6. Visual cues supplementing verbal material, particularly in tasks where the child is required to remember key information, enhanced game appeal and useability. We found that simply presenting problems verbally was not nearly as effective as combining this material with redundant onscreen information. Children drew on both modalities in recalling the information they needed, and used the visual cues as a reminder when they accidently forgot or did not attend to the verbal material.

Conclusions

The use of speech in interactive products constitutes a new area for educational psychologists and teachers. The findings from the present studies suggest several concrete criteria that can be used to determine the effectiveness of speech as a part of preschool software. They are applicable to all interactive media, not just computer software; any multimedia technology (videodiscs, interactive compact discs, etc.) intended for preschool use that employs spoken language can be evaluated using the following guidelines.

1. The length of the spoken segments in the program should be brief. Long, expository segments tax young children's listening abilities and attention spans.

2. The language used in the program should be as concrete as possible, and should direct the child to undertake a specific action. For example, if the task is to match animals with their appropriate sound, an ambiguous error phrase such as "Try it again," should be avoided; a phrase like "Pick a new animal," is better.

3. The speech should be as natural as possible, using appropriate inflection wherever possible. Positive affect in messages such as "That's right!" and neutral or questioning affect on wrong answers seem to enhance children's ability to understand the program, as well as help motivate task performance.

4. Programs employing a varied number of messages for the same situation are preferable to those with only one, repeated message. In our studies, children quickly tired of repeated phrases and came to ignore them, defeating their purpose.

5. The language in the program should be supplemented by visual material. The speech segments should have a concrete relationship to objects or events happening on the screen. Such redundancy between spoken and visual elements helps children to "figure out" what to do, if they do not fully grasp the verbal material. In our math game, for example, the verbal problems are actually performed as they are verbally described. This allows children who may initially fail to process the spoken material to nonetheless play along, and it also encourages children to connect the language used to the situation at hand.

References

Ackerman, B. (1981). Performative bias in children's interpretations of ambiguous referential communications. *Child Development, 52*, 1224-1230.

Beal, C. R. & Belarad, S. C. (1990). The development of message evaluation skills in young children. *Child Development, 61*, 705-712.

Hargrove, P. M. & Panagos, J. M. (1982). Role saliency cues and children's sentence comprehension. *Journal of Psycholinguistic Research, 11*, 217-228.

Markman, E.M. (1979). Realizing that you don't understand: A preliminary investigation. Child Development, 48, 986-992.

Parham, C. (1988). Computers that talk. Classroom Computer Learning, 8(6), 26-31.

Reitsma, P. (1988). Effects of guided reading-while-listening and independent reading with computer-based speech feedback. *Reading Research Quarterly, 23*(2), 219-235.

Shatz, M. (1978). On the development of communicative understandings: An early strategy for interpreting and responding to messages. *Cognitive Psychology, 10*, 271-301.

Wiener, R. (1991). Computerized speech: A study of its effect on learning. *T.H.E. Journal, 18*(7), 100-102.

Special Presentation
Software —From Research to Development to Publication

Barbara Kurshan, Chair
Educorp Consultants, Roanoke, VA

Software development for education is occurring at a variety of levels and at a dramatic pace. As the market demands more computer and multimedia instructional products, the consumer and the developer has to be able to determine if the product is "valuable" and "viable." A recent education study group has defined software of value when it is used consistently and it can be employed as a vital element to improve teaching. While software is viable if it survives in use long enough to enable courses to be changed by the software. This panel will address the process of researching and developing valuable software that publishers will think is viable. The presenters will explore the future of software in the K-12 and collaborative higher education market from the perspective of funder, developer, and publisher. In addition, the session will address issues such as how to approach funding sources, working with developers in your classroom and taking your product to a publisher. Topics covered will include:

- Obtaining funding for software development research
- Current research based software projects
- New software and multimedia development
- How to get development projects published
- The next generation of software

Panelists:

Marge Cappo
Wings for Learning, Inc.
Scotts Valley, CA

Steve Ehrmann
Program Office for Interactive Technologies
Annenberg/CPB Project,Washington, D.C.

Peter Kelman
Scholastic Inc.
New York, NY

John Richards
Software Development Manger,
BBN Laboratories Incorporated
Cambridge, MA

Special Presentation—ISTE
Hypermedia in the Classroom

Ray Braswell
Auburn University, Alabama

Mark Horney
University of Oregon

This session will discuss and display classroom materials and projects developed by teachers and researchers in the area of hypertext, hypermedia, and multimedia.

Projects
History/Social Studies and Hypermedia/Multimedia or Interactive Media

Using Hypermedia to Implement Curricular Change

Linda Smith, District Technology Specialist
Sheri Henderson, Mentor
Saddleback Valley Unified School District, California

Abstract

After 18 months, Saddleback Valley Unified School District's Hypermedia Project has resulted in teachers authoring 65 original Hypermedia stacks in at least 13 areas of curricular reform. Hypermedia is the linking of media via the computer. This school district of 25,000 students in Mission Viejo, CA specifically chose to link the laser videodisc player to the Apple Macintosh computer using *HyperCard* software. Thirty-nine teachers from 29 schools were given the training and resources necessary to include Hypermedia in the curricular and instructional reform process.

Sheri Henderson, one of our presenters, will demonstrate her simple yet powerful *HyperCard* utility program which allows anyone to create interactive, multimedia lessons using any videodisc; users merely copy, paste, and type. Examples from her project—which integrates American literature, art, and history—will show its ease of use and flexibility.

This Hypermedia presentation was produced through the Project to share our successful strategies for districtwide implementation of multimedia in all subject areas. Attendees will receive handouts on selecting participants, training, budget, follow-up, increasing publicity opportunities, purchasing, and recommendations for multimedia software and hardware.

Civil War Interactive Multimedia Project

Charles White
George Mason University, Virginia

Abstract

The proposed project presentation will report on the status of the Civil War Interactive Multimedia Project, a design and development project currently funded by the Corporation for Public Broadcasting and being carried out by the Center for Interactive Educational Technology at George Mason University in Fairfax, Virginia.

The project is based on the PBS series The Civil War, a documentary by Ken Burns of Florentine Films, Aired on five consecutive nights in September of 1990 Burns' series achieved the highest ratings in the history of public broadcasting. The task facing the interactive multimedia designers has been to capture the power of the video material while broadening and deepening the learning opportunities through the power of an interactive multimedia environment. The presentation will focus on four key design challenges and how we have met those challenges.

The first challenge has been to design a product that will address some of the serious deficiencies in social studies knowledge and critical thinking skills reported in recent national assesments and in the reform literature. Knowledge of the Civil War is marginal among high-school seniors; indeed, some 68 percent of them could not place the Civil War in the Correct 50 year period. knowledge of fundamental principles of government underlying the war are also not well understood by today's students. Based on extensive examination of the literature, we derived general principles of history, key historical themes, and knowledge, skill, and value goals for the project that we believe target some of the most glaring gaps in student understanding.

Another challenge has been to glean the most useful and effective video, audio, and textual data from Ken Burns' considerable archives. Burns had a 14-to-1 shooting ratio for his 11-hour broadcast series, yielding some 150 hours of video material to be reduced to two sides of a videodisc. Thousands of still photographs and hundreds of pages of diaries and documents have been at our disposal, and we will describe the procedures we adopted to manage the selection task.

A third challenge has been to provide maximum flexibility for exploration while providing sufficient structure to guide student inquiry in a rich knowledge base. the designers believe that students with a wide range of learning and cognitive styles should be successful users of interactive multimedia. We will discuss some of the difficult trade-offs required to achieve what be believe is an appropriate balance of free-wheeling investigation and guided inquiry.

Finally, the design team has been focusing on teacher training throughout the project. We will discuss a multi-dimensional strategy for teacher training built into the design of the videodisc and print materials as well as ancillary video-and print based training materials.

Given the limited amount of presentation time, detailed information relating to our discussion will be delivered via handouts.

Teaching History/Social Science Through Interactive Technologies

Nancy Hunt
California State University, Fresno, CA

Abstract

We live in a media-rich society, surrounded by television, radios, movies, and digital sound. Satellite communication makes possible the local publication of a nationwide newspaper and allows us to see and hear events as they happen around the world.

If we are to make school an exciting and relevant place to be, we must reach our students through, and empower our students to use, the technological tools of the age. Unfortunately, the typical classroom is less well equipped than the average American den.

This presentation will describe a project which addresses this technology-gap between the typical home and school environment. In the fall of 1990, faculty from CSU, Fresno and the Clovis Unified School District were awarded an IBM equipment grant to develop history/social science lessons and staff development materials which use interactive laserdisc technology.

This presentation discusses the process by which the teachers were introduced to laserdisc technology and assisted as they created their interactive lessons. Lesson excerpts will be presented and any changes evidenced in teacher and student attitudes towards the content area and the use of technology in the classroom will be discussed.

Special Presentation

Exploring Simulations in Science

John Richards
Bolt Beranek and Newman

Yoel Givol
Logal Educational Software

Bill Barowy
Maine Maritime Academy

Mark Rodriquez
Lexington High School, Lexington, Massachusetts

Simulations bridge the gap between complex mathematical theories and experience, and are changing the very nature of scientific inquiry. They provide new visual phenomena that aid in building scientific intuitions. Scientists are developing new, investigative skills, and asking novel, searching, questions.

Science teaching can now capture these exciting developments and reflect the way scientists presently work. Understanding processes is easier when students can explore the dynamics and nature of the changes made. Students can further their knowledge when they are given the opportunity to become actively involved in altering a process, and predicting and observing the effects of their modifications. by questions and the skills to seek informative answers.

This panel will discuss their experiences with the Explorer system: an interactive environment for exploring simulations that gives students and teachers access to new tools that are designed to provide students with these experiences. The Explorer provides.

- Simulations: taking advantage of the computer's ability to present interesting, interactive animation. Dynamic, graphic simulations in physics (mechanics, light, waves, electricity, and modern physics) and biology (genetics, heart, population, photosynthesis).

- Analytical tools: providing access to multiple related representations of the data. The spreadsheet, graphs, histograms, and numerical displays update in real time while the simulation runs. Graphs are drawn or data sampled to the spreadsheet as you view the simulation

- Scripting: provides facilities to design labs, experiments, and lessons. For example, you might create a button for graphing the energy of a system as a function of time, or write a script defining a condition to detect a collision.

Special Presentation—ACM/SIGCUE

Cooperative Learning - Computers and Technology

Greg Brownell, Nancy Brownell
Bowling Green State University

This session will be an overview of cooperative learning strategies and methods for integrating cooperative learning into computer education. A report will be given of a research study on cooperative learning.

Practical implications for teachers will be discussed, and materials will be made available.

Projects
Teachers and Telecommunications

CAWP Online: Supporting Professional Growth with Electronic Linkages

Jane A. Rosen
Instructional Coordinator Pritzker School
Chicago Board of Education

Abstract

The presentation will discuss a project which electronically links Chicago Area Writing Project's summer institute with each other and the Writing Project Director. CPS installed an EIES network in September, 1989 but it was initially underutilized by local educators and was not directly tied to district curriculum and instruction objectives. The investigators connected participants from the well established Chicago Area Writing Project with the exciting Bulletin board System as a economical effective, and efficient means of facilitating greater support for and interaction between Writing Project teachers, a secondary goal was to publicize the project and gain support for expansion. Moreover, they intended to apply the model they developed for teachers of writing to other curricular purposes.

In this session, the presenters will describe the following aspects of the project in the handouts and remarks during the session:

- the process of obtaining support, approval, and assistance from CPS central office and building administrators;
- procedures for initial on-line training of Writing Project teachers
- the on-going technical and curriculum support system provided to Writing Project teachers
- preliminary findings on the effectiveness of this project
- efforts to publicize and gain support for the project within the Chicago Public Schools
- future plans to refine and expand this program and to extend the concepts to support and sustain staff development in other curricular areas

Teacher Inservice Technology Education:
A Partnership Approach in California.

Craig Blurton
Director, California Technology Project
Seal Beach, CA

Abstract

The California Technology Project is a cooperative effort of the California State University system and the California Department of Education to provide support services for technology-using K-12 and preservice teachers educators. This session will focus on the free statewide telecommunications network, the Technology Resources In Education (TRIE) electronic telecommunications network, the Technology Resources In Education (TRIE) electronic information service, maintained by the project for use by educators.

Special Presentation

Virtual Realities: From the Concrete to the Barely Imaginable

Stephen Marcus, Graduate School of Education
University of California, Santa Barbara

This session deals with "deep media". If text *tells*, and video and film *show*, then virtual realities *embody* the worlds they create. One seems to enter such simulations physically. These environments involve unusual interfaces and metaphors for future educational computing. Included are systems like *Reality Built for Two*, *the Virtual Environment Workstation*, *Mandala*, the *Alternate Reality Kit*, and *A Very Nervous System*. The presentation includes a discussion of virtual reality hardware and software and explores the educational implications of radical shifts in point of view (e.g., one can *become* the data), of being responsible for what someone else sees and of functioning in environments in which space itself is the interface.

Projects
HyperTraining

HyperMedia: A Performance-Based Model of Inservice Education

Marty Karlin
Trinity Center, California

Abstract

The State of California Curriculum Frameworks are calling for teachers to incorporate active, thematic, and cooperative learning methodologies across the curriculum. Students are to learn to obtain and value knowledge, reach conclusions and develop sound judgement. This inservice project seeks to make the training, tools, materials, and support available to help teachers and students meet those requirements.

As 30 teachers in 17 rural California schools learn to access and employ interactive multimedia technologies in their classrooms, they become exposed to Information Age teaching and learning environments that are intuitive and versatile, and have great appeal. Critical factors have been administrative support, curricular integration, equipment availability, ease of use, training, and individualized follow-up. This report will include a description of the rationale, goals, expectations, training activities, and results after two years of acquisitions and inservice. Teaching methodologies, effects on learning, and teacher/student products will also be discussed.

HyperTEACH- A *HyperCard* Teacher Utility

Michael Land
Midwestern State University, Texas

Abstract

In an effort to get preservice and inservice teachers to begin to use *HyperCard* to develop instructional stacks with a minimal amount of training, the presenter prepared a teacher utility stack called HyperTEACH. Students can begin to prepare their own stacks after only a few hours of practice with HyperTEACH. With HyperTEACH, an individual can prepare an instructional stack that randomly displays some type of encouragement after correct responses, and indicates when the user has selected incorrect responses, and returns them to the original question. The presenter will share some of the stacks his students have prepared using HyperTEACH. Individuals at the session will be given copies of HyperTEACH that they can use with their students.

HyperCard for Elementary and Secondary Education: A Sampler

Kevin Walsh
Connie Gonser
John H. Strange

Abstract

For the past two summers the school district of Royal Oak (Michigan) has conducted a one week, 40 hour course in HyperCard for teachers in the district. almost half of the teachers who attended the HyperCard training had never used a computer before. All the teachers produced products, many of which have been or could be, developed into instructional programs for use within the district. This session will enable participants to see a sample of the teacher produced products, to receive a diskette of that sampler, and to discuss the strategies that we have followed in encouraging additional teacher training throughout the year.

Special Presentation

Two International Telecommunication Models: I*EARN (The International Education and Resource Network) and the New York State/Moscow School Telecommunications Project

I*EARN was launched by the Copen Family Fund, Inc. in 1989. It is a pilot model project that is exploring the impact of telecommunications internationally on teaching methodologies, student learning and international understanding. The purpose of the telecommunications is to have students work together, internationally, on projects which they design - projects which will, in some way, make a meaningful contribution to people and the planet. I*EARN is an expansion and international application of the New York State/Moscow School Telecommunications Project.

At this writing, there are 5 Centers in the US that can act as regional models for the demonstration and replication of this work. The Centers are:

1. The GEM Project in Yorktown Heights, New York.
2. The ELNA Project in Boston, Massachusetts.
3. The Century 21 Project in Seattle, Washington.
4. The Irvine School District - as part of the Education for the Future Project in Irvine, California.
5. The Albuquerque Public Schools.

The five US Centers of I*EARN are connected by telecommunications to the USSR (Leningrad, Moscow, and Kiev), Israel (Jewish and Arab Schools), the Netherlands and Argentina. In the next few months it is expected that schools in China (PRC), Spain, the Phillipines, Indonesia, Finland and England will join the network. Discussions are in process with schools in Mexico, Costa Rica, Egypt, France, and Japan.

The New York State/Moscow School Telecommunications Project is the first of its kind between the US and the USSR. In June 1988 an agreement was signed between the Copen Family Foundation (working in collaboration with the NYS Education Dept.) and the Soviet Academy of Sciences to initiate a 3 year pilot project to assess the impact of telecommunications and exchanges on teaching methodologies, student learning and international understanding.

This project links 13 schools in Moscow with 13 schools in New York State by telecommunications - electronic mail and video speaker telephones. Student, teacher and administrator exchanges support the relationships between the schools.

The purpose of the school linkups by telecommunications is to have students engage in projects of their own design - projects which can in some way make a contribution and be of service to people and the planet while meeting academic requirements. The purpose of the exchanges is to further support the student generated projects as well as increase international understanding.

At this writing, several hundred students, teachers and administrators have participated in telecommunication projects and exchanges (home stays, primarily) and produced some remarkable results.

Since the purpose of this work is to create a model for replication by others, the Project presentation will focus on a brief summary of the work, highlights of the successes, an articulation of the problems and an open discussion of issues about replication.

Panelists:

Peter Copen
The Copen Family Fund, Inc.
Yorktown Heights, New York

Ed Gragert
The Copen Family Fund, Inc.
Yorktown Heights, New York

Alexander Uvarov
The Soviet Academy of Sciences

Jane Bailey
NYSED

Paper

Ending the "Blind Leading Blind" Syndrome: Advocating Technologically Competent Administrators

John F. Beaver
SUNY College at Buffalo, New York

The late 1980s witnessed the rise of many crucial social issues that captured the national spotlight and cried loudly for immediate solutions. An increased emphasis on drug education, prevention of teen pregnancy, AIDS awareness, and other current concerns are causing school administrators to shift their attention, and their funds, from instructional computing to other programs. As Mary Jo Langhorne, et al, suggest: "The use of computers in education is not the compelling issue it was five—or even two—years ago. New bandwagons have presented themselves: Higher order thinking skills, merit pay plans, and objective-based education now compete for the attention and funding computers received but a short time ago."[5] Although educational computing has received strong public support in the recent past, many new topics are monopolizing media attention.

During the past three years, many schools have had fewer resources available to them. When "belt tightening" takes place, some programs receive insufficient support. How have instructional computing programs fared in this environment of financial instability? In 1987, I conducted a national study of outstanding elementary school instructional computing programs. One disturbing finding was that the schools' computer budgets were very volatile, with few schools having established computer expenses as permanent line items in their school budgets. Kathleen Fulton reports that the Office of Technical Assistance found similar tendencies: State-level profiles indicate that state support is subject to the volatility of economic conditions, and of political and educational priorities in the individual states. Another pattern to emerge from the Beaver study (1987) was that funding levels for instructional computing programs had declined dramatically during the late 1980s. The findings indicate that many administrators felt they'd already paid their dues by purchasing some necessary hardware. They were now prepared to allow other important educational issues to take financial precedence. For the most part, they lacked long-range plans to guide the steady development of their programs.[1]

In addition to the fiscal instabilities, some sources continue to challenge the importance of using computers educationally. A recent *New York Times* headline states: "Computers are in the classroom, but no one is paying much attention to them." A headline in the *Wall Street Journal* reads: "Computers Failing as Teaching Aids." The popular media suggest a reduction of interest in educational computing at a time when more support is needed to ensure maintenance of programs. Langhorne, et al, point out, "...there are those who are already concerned about the waning interest in instructional computing and the decline in financial support for microcomputers." This confused situation promises programmatic chaos unless school leaders provide guidance and support. The leaders must develop clear understanding of instructional computing program needs so that they can develop long-range plans that insure those needs are met. Unfortunately, too few school leaders have the necessary computer competence, and consequently, far too few comprehensive plans exist.

A recent National School Board Association survey of American school districts concluded that there is very little evidence of comprehensive planning for educational computing programs. The survey findings conclude that most district plans "did not have a comprehensive flavor." Why is this true? Developing a comprehensive, long-range plan that will support an educational computing program requires a leader with a deep understanding of the problems and promises of instructional computing. *Many* of our administrators simply do not have this understanding. Greg Kearsley contends that 'administrators often lack any suitable preparation to effectivly manage the computing activities in their schools."[4] I agree our administrators are being asked to make decisions on a topic for which they are not prepared.

If we expect our administrators to provide the vision and understanding needed to guide the development of instructional computing programs, we must encourage them to increase their computer competence. As Sylvia Charp argues: "It is generally agreed that more still needs to be done in faculty development if we expect truly broad use of technology in teaching and learning to occur. This should include programs designed for administrators of all levels, to enhance the awareness of technological literacy and their competence in planning for technology utilization."[2] The administrative programs Charp references should not be limited to inservice experiences, but should include college level instruction that constitutes a portion of the required program for state administrative certification.

Kathleen Fulton points out that only "...18 states require...that all students in their teaching degree programs or those seeking certification take a course on [educational] computer topics."[3] About one-third of our states value educational computing enough to mandate it in their teacher education programs. And, if less than one-third of the states require computer education for teacher certification, far fewer require a similar course designed for administrators. Still, without such experience to help administrators develop the computer competencies they need, there will be no clear vision guiding instructional computing programs.

Many educational leaders have called for greatly increased computer training for our teachers. For instance, as Robert McCarthy quotes Tom Snyder: "We all made a serious mistake in giving the computers to the kids first. We ignored the teachers, cut them out of the loop, and that's precisely the way to kill a promising educational technology."[9] Len Scrogan includes other limiting factors, but agrees with Snyder's main point: "Why isn't technology having a greater impact? Many factors are to blame: insufficient access to computers, inadequate supplies of quality software, and pauper technology budgets all contribute....[but] the most significant factor lies in the fact that the vast majority of teachers today have little or no training in the use of the new technologies."[11]

In general, even teachers are still not receiving the preparation they need to use technology successfully. Dave Moursund points out: "By and large our colleges of education are doing a miserable job of preparing teachers to deal with the Information Age."[10] Granted, teachers *do* need more assistance with using technology instructionally. But who is in the best position to insure that the necessary logistical, financial, and psychological support that teachers need is provided for them? Their local administrators, of course, if the administrators understand the unique need for support that instructional computing adoption requires.

Certainly, empowering our teachers—raising their level of technological competence—is very important. I strongly endorse greatly increased staff development efforts to raise teachers' technological understanding and comfort level. However, if the promises of educational technology are to be kept, the education of our administrators is equally—perhaps more—important. Somehow, when we started the computer revolution we forgot to invite the administrators .

Too often, unfortunately, administrators have no hands-on experience or quite limited hands-on experience with microcomputer experience. Without that personal experience, they can't really understand the need for continuing support that instructional computing demands. Administrators have to develop the experiential base they'll need to guide their instructional computing programs. One part of that foundation includes the hands-on experiences that a course on administrative uses of microcomputers can provide.

Computer educator David D. Thornburg argues strongly for the need for informed leadership: "Educational computing took its sabbatical last year, and now it's time to get to work, recharged and reinvigorated with energy and enthusiasm. Yes, additional hardware costs more money, but that's not the problem. The problem, as I see it, comes from a lack of vision."[12] Although I agree with Thornburg, I suggest that the road to vision includes journeys through the "slough of frustration"—lost files, confusing screen displays, unclear support documentation, printer malfunctions, hardware and software incompatibilities, information overload—to the "highlands of empowerment"—polished document production; facile data manipulation, swift numerical analysis, powerful instructional delivery. If administrators have no personal experience with microcomputers, they can't really understand the training and support that teachers need to help them learn to master the machines. Without that understanding, developing educational computing vision is not possible.

These points were brought home sharply to me this summer when I taught an educational administration course titled "Administrative Uses of Microcomputers" at SUNY College at Buffalo. My experiences parallel those reported by Kearsley when he taught a similar course three thousand miles away. "Many of the administrators who take the course have had no prior experience with computers and in some cases are antagonistic or anxious about them."

The course I taught is an elective in the educational administration certification program. Course participants are aspiring administrators, individuals seeking administrative certification as a precursor to finding an administrative position. Nearly half of those enrolled in the two sections had *no* previous computer experience. None! Some were not even familiar with a keyboard.

Most of those without experience shared another characteristic: they were afraid of computers. My experience supports Andrew Barbour's findings from a national survey of administrators: "17% of the administrators cite fear of computers among their staffs as a major problem facing their wider implementation in their school or district."[1] Although Barbour doesn't mention it, I suspect many administrators share the fear of technology with their staffs. That fear is a fear of the unknown. And that fear, as with all fears of the strange or unusual, dissipates with understanding and familiarity. Those qualities only come with experience.

The course proved especially rigorous for those starting from scratch, designed as it was to provide a broad range of experiences and insights to participants. However, after completing the course, the participant consensus was that the hands-on expectations provided a critical part of their experience. If they hadn't had the opportunity to master the computer for themselves, they felt that they would not have understood the powers and the problems of technology adoption. When they eventually receive administrative appointments, these leaders-of-the-future will not be guilty of "dumping" hardware on untrained teachers. They understand the need for continued, substantial support, both financial and psychological.

The course includes practical applications, group discussions, individual and co-operative learning experiences, and opportunities to explore in depth a particular application of choice. The applications include the productivity enhancement standards: word processing, data base managing, print merging, spreadsheet developing, chart generating, information interchanging between applications. The power of each application is discussed, with special attention given to how that power applies to the role of an educational administrator.

Each application is also taught during class sessions, using Microsoft's *PC Works* because of its power, ease of use, and flexibility. Next, students practice using the application independently while completing assignments which represent typical administrative uses of that application. For instance, once they learn word processing, students are asked to produce an "opening-of-the-school-year" letter that demonstrates appropriate use of the powers provided by word processing. The powers mastered include changing font style, font size, font type, spacing, and justification.

After data base management is taught, participants practice it by creating a student information file. Then they demonstrate using the data base program's searching, sorting, and reporting powers by printing out reports appropriate for different situations. The next computer use introduced is print merging—combining word processing and data base powers. Students demonstrate their print merging proficiencies by generating personalized form letters that satisfy on-going administrative communication needs such as parent open house communication, parent conference letters, or field trip announcements.

Once spreadsheets have been mastered, class members use them to create budgets or inventories for their schools. Next, they use the figures from their spreadsheet budgets to generate graphic forms using *Works*' charting powers. Finally, they combine their understanding of word processing, data bases, spreadsheets, and charting by sharing data from one application component to another. These integration powers let students develop polished word processed reports that include spreadsheet figures, data base records, and charts.

In addition to the practical applications that focus the course, class participants also engage in varied group discussions and problem solving experiences. These situations provide more pervasive opportunities for applying their computer understandings. Key topics included in course discussions are deciding administrative computer uses, establishing optimal computer adoption approaches, creating stable computer budgets, determining computer placement, planning instructional computing programs, planning staff development of computer competencies, planning computer integration policies, and developing cost-effective policies for computer usage.

Beyond the course's individual and class requirements, each participant is asked to work in a cooperative partnership. This task demands a joint approach to learning a new technology and presenting the pertinent issues it introduces to the rest of the group. Each individual forms a dyad with another class member. Each pair is expected to become familiar with a "new technology" topic of agreed-upon choice and report on its administrative implications to the class-at-large. Suggested subjects that the teams are encouraged to investigate and report on include applications of telecommunications, uses of dedicated input devices (such as scanners and digitizers), the costs and features provided by dedicated applications (like scheduling and attendance programs), the administrative implications of new technologies (including CD-ROM and interactive laser disks), and desktop publishing (and desktop presenting) applications. These partnership explorations of new technologies help administrators recall the time needed to master each new innovation. That understanding will prove useful to them later when they plan and implement staff development projects designed to encourage the adoption of new technologies.

Each of the elements of the administrative computer competence course expectations are important. Without the hands-on experience, discussions become remote, second-hand experiences. Without the discussions, the hands-on experiences degenerate to software training workshops. Without the discussions and hands-on workshops, the project presentations become "Show-And -Tell" sessions. Together, the three components provide a solid foundation for an administrative computer leadership capable of guiding us steadily into the next decade. The course prepares administrators who can and will model appropriate technology uses in their own school classrooms and offices.

Langhorne, et al, voice a major concern of educational computing advocates: "Educators have an embarrassing history of adopting the latest fad, trying to make those fads meet all needs, becoming dissatisfied because the new idea did not bring success in every application, and so abandoning the innovation and moving on to the next." We don't want the computer to share the fate of instructional television and other educational innovations that came and went. Robert McCarthy expresses this concern well: "The computer is too powerful an educational tool to be cavalierly consigned to the educational broom closet."

If we are to keep the computer out of McCarthy's broom closet, we will need the vision and leadership of our school administrators. Our educational leaders must establish long range plans to guide their instructional computing programs. Daniel E. Kinnaman reports the recommendations of the National Education Association's committee on educational technology: "All schools should develop and implement a plan to install a computer with adequate software on the desk of each teacher by 1991. Resources for educational technology must be adequate to provide balanced support for staff development, software, hardware, and research into curriculum integration and development.'" To develop such plans we must have informed, knowledgeable, and dedicated planners—planners who have learned the powers and pitfalls of educational technology adoption. The only way to avoid the computer broom closet is to require our administrators to raise their computer competence. That task can best be accomplished with instruction dedicated to that purpose. A course like the one described here, or a similar one in a different format, should be a required as part of the preparation for receiving and maintaining administrative certification. Our educational leaders must develop a clear understanding of what they are expected to lead.

References footnoted in text

1. Barbour, Andrew. (April, 1987). "Office romance: Why administrators are hooked on technology." *Electronic Learning.*
2. Beaver, John F. (April, 1987). *A national search for quality: An examination of high-quality elementary school instructional computing programs.* Unpublished dissertation, Michigan State University.
3. Bruder, Isabelle. (June, 1989). "NSBA study says largest districts have weak long-range technology plans." *Electronic Learning.*
4. Charp, Sylvia. (June, 1989). Editorial. *Electronic Learning.*
5. Fulton, Kathleen. (October, 1988). "Preservice and Inservice: What Must Be Done in Both." *Electronic Learning.*
6. Kearsley, Greg (November, 1988). "What Should Today's Administrators Know About Computers?" *Technology Horizons in Higher Education Journal.*
7. Kinnaman, Daniel E. (September, 1989). "NEA committee on educational technology releases report." *Classroom Computer Learning.*
8. Langhorne, Mary Jo, Jean O. Donham, June F. Gross, Denise Rehmke. (1989). *Teaching with computers A new menu for the '90s.* Phoenix: Oryx
9. McCarthy, Robert. (September, 1988) "Making the future work: The road to computer integration." *Electronic Learning.*
10. Moursund, Dave. (June, 1989) "Why are our colleges of education continuing to graduate computer illiterate teachers?" *The Computing Teacher.*
11. Scrogan, Len. (January, 1989). "The OTA report: Teachers, trainers, and technology." *Classroom Computer Learning..*
12. Thornburg, David D. (October, 1988). "A time to take stock of the state of educational computing." *A+ Magazine..*

Paper

Selecting Personal Computers for a University MicroComputer Cluster: A Multiple Criteria Decision Analysis Approach

Philipp A. Djang
U.S. Army Training and Doctrine Analysis Command
White Sands Missile Range

Abstract

This paper describes a Multiple Criteria Decision Analysis Approach (MCDAA) to determine the selection of personal computers for a university personal computer cluster. The MCDAA combines the capabilities of Analytic Hierarchy Process (AHP) and Integer Goal Programming (IGP). Decision makers can use this approach to determine "what kind" of personal computers and "how many" of each type of personal computers could be purchased. A real world application of the MCDAA illustrates how the processes are combined to answer the "what kind" and "how many" questions.

Introduction

A university computer center is a dynamic organization that reflects the state of computing technology. The pressure to provide up-to-date educational computing facilities to the students, faculty and staff present a uniquely significant problem. As university computer centers adopt personal computers into their array of technology and develop the facilities and personnel staffing that are required to operate a personal computer cluster, two significant decisions confront them. These decision are to determine "what kind" of personal computers should be purchased and, given a limited budget, "how many" of each kind of personal computer can be purchased.

The purchase of a particular type of personal computer for a microcluster is based upon many criteria. The principal criterion for the selection of a personal computer in a university setting is based upon its educational value. Additional criteria are cost, maintenance, compatibility with current and future work environments, network compatibility and popularity with students. Since this is a significant capital outlay, it is important that a center select those computers that are consistent with the educational goals of the university.

A Multiple Criteria Decision Analysis Approach

Personal computer purchasing decisions can be expedited by an application of an integrated multiple criteria decision analysis approach. The integrated Multiple Criteria Decision Analysis Approach (MCDAA) combines the capabilities of Analytic Hierarchy Process (AHP) and Integer Goal Programming (IGP). The AHP can assist decision makers with the resolution of complex problems that involve multiple criteria [4]. Previously, the AHP has been used to select multi-micro distributed accounting information systems [1] and evaluate micro/mini computer configurations [2]. This paper extends these applications by describing a method for combining AHP and IGP so that the purchasing decision is founded upon both the subjective values of a decision maker and the objective constraints of money and space.

In this approach, decision makers identify, for each criteria, the relative preference for each of the decision alternatives by an application of the AHP. The result of the process is a prioritized ranking of each of the decision alternatives with respect to the perceptions of the decision maker. Each of the decision alternatives receives a ranked score on each of the criteria. The results quantify the decision maker's preference for each decision alternative and provide a means for answering the "type of computer" question. In order to determine "how many" personal computers to purchase, IGP can be used. IGP is a linear optimization method that incorporates the preferences of the decision maker [3]. The priority ranking for each type of alternative can be incorporated into an IGP. The AHP rankings provide an efficient means for determining the priority weights for the IGP.

The Analytic Hierarchy Process

The AHP is appropriate when the criteria cannot be accurately measured or the decision alternatives cannot be precisely differentiated. This particular methodology is used because one is basically interested in a value assessment of the deterministic factors rather than their utility [1]. The primary advantage of the AHP is that it provides a methodology that includes both objective, measurable criteria and the subjective perceptions of the decision makers. The methodology of the AHP requires that a hierarchical structure be developed for the decision. The hierarchical structure consists of three levels. The levels are the overall goal, the criteria and the decision alternatives.

The overall goal for this application will be to select the type of personal computers (decision alternatives) that satisfy the criteria. The decision alternatives for this application are the

IBM PS/2, an IBM compatible and the Apple MacIntosh. Additional brands of personal computers could be easily considered in the decision making process. The analytic hierarchy for a university microcluster is given in figure one.

For this application of the AHP, the criteria were divided into two classes, an educational class and a functional class. The educational class consisted of the criteria that could be classified as falling in the educational domain, while the functional class was composed of the criteria that dealt with daily operations. A total of fifteen criteria were generated. Five of the criteria fell into the educational class and the remaining ten criteria were classified as functional criteria. The decision maker was asked to compare the alternatives on each of the criteria.

The AHP methodology is based upon the construction and analysis of a pair-wise comparison matrix [4]. For each decision criteria, the decision maker compares two of the decision alternatives on the criteria and then ranks the comparison according to the scale of values (9=high preference, 1=equal preference). For each criteria, n(n-1)/2 comparisons were required. A positive matrix was constructed which consists of the values and their reciprocal of the values. The reciprocal values were entered into the transpose position of the square matrix. All decision criteria matrices are square, since all decision alternatives are compared. The diagonal of all the decision matrices are given the value of one (consistent with the rating scale). The pairwise comparison matrix is normalized by computing the sum of each row and dividing each element by the sum. A principal vector is constructed by averaging the normalized row values. For the criteria, the principal vector contains the relative priorities for each decision alternative.

The consistency of the rankings can be evaluated by a few matrix operations. The pairwise comparison matrix is multiplied by the principal vector. From the resulting matrix, the sum of the row means are divided by the principal vector. The result vector is summed and divided by the number of decision alternatives. The result, λ_{max}, is used in computing the consistency index (CI).

$$CI = \frac{\lambda_{max} - n}{n - 1} \qquad \text{(1) Consistency Index}$$

The consistency ratio (CR) is computed by dividing the CI from a value obtained from a table of random consistency. If the CR is less than 0.10, the comparison exhibits internal agreement. Otherwise, the criteria would require additional evaluation.

The Analytic Hierarchy Process produces a prioritized evaluation of each of the decision alternatives. Each of the decision alternatives (IBM, IBM compatible and Apple MacIntosh) receives a score on each of the criteria. The criteria scores are combined into an overall score. The overall score indicates the relative importance of each personal computer and assists the decision maker in determining where resources should be allocated. The results of the Analytic Hierarchy Process quantifies the decision maker's preferences for each decision alternative and provides a means of determining the "what type of computer to buy" question. In order to determine "how many" personal computers to purchase, integer goal programming can be used.

Goal: Select Microcluster Personal Computers

Educational Class	Functional Class
Student Preference	Purchase Costs Repair Cost
Homework Compatible	Space Limits Staff Skills
Ability to Operate Educational Software	Availability of Peripherals Compatible w/ Network Ease of Upgrades
Faculty Preference	Compatible w/ Software Vendor Relationship
Future Computing Environment of Employer	Vendor Market Standing
Decision Alternatives:	1) IBM PS/2 2) IBM Compatible 3) Apple MacIntosh

Figure 1. Analytic Hierarchy for a University Micro-Cluster

Integer Goal Programming

The priority ranking for each type of personal computer can be incorporated into a goal programming model. The AHP rankings provide an efficient means for determining the priority weights for a integer goal programming model. Goal programming is a linear programming method that is appropriate when an organization has multiple conflicting objectives and is attempting to satisfy all of the objectives under the constraints of scarce resources. Goal programming has the capabilities to analyze the decision maker's multiple aspiration levels, to relax some of the linear programming constraints and to incorporate the decision maker's preference system for multiple conflicting goals. Goal programming is a method for transforming an optimizing model into a satisficing model by incorporating the decision maker's preferences. [3] In goal programming, all of the objectives are assigned target levels (the goals) for achievement and a relative priority is placed upon achieving these levels. Goal programming attempts to find an optimal solution that comes as "close as possible" to the targets in the order of the specified priorities. [3]

A small integer goal programming model was developed to determine the quantity of personal computers that the university microcluster should purchase. The priorities for the goals were obtained from the AHP. A hypothetical funding level was used as a limiting constraint on the number of computers, network accessories and software could be purchased. Estimates of the actual physical space, personal computer, peripheral and network costs were obtained by interview with the Director of a university microcluster. The goal of the integer goal programming model is to satisfy the AHP priority values and at the same time, minimize the shortfall from the goal of purchasing an integer number of computers and associated peripherals, subject to funding and physical space constraints.

Pairwise Comparison Scale

Verbal Judgement of Preference	*Rating Scale*
extremely preferred	9
extreme to very strongly preferred	8
very strongly preferred	7
very strongly to strongly preferred	6
strongly preferred	5
strongly to moderately preferred	4
moderately preferred	3
moderately to equally preferred	2
equally preferred	1

Figure 2. Rating Scale

An Example

A structured interview/questionnaire was developed to elicit the priority of the criteria of the different computers. The director of a microcluster was interviewed and the rankings for each of the criteria were collected and analyzed according to the methods prescribed by the AHP. The results of applying the AHP produced a prioritized evaluation of each of the decision alternatives. The rating scale is given in figure two. For one of the criteria, the raw preferences (left side choice preferred over right) that the Director expressed were:

IBM PS/2 versus IBM Compatible = 3.0
IBM PS/2 versus MacIntosh = 3.0
MacIntosh versus IBM Compatible = 3.0

From these data, the square pairwise comparison matrix is constructed. The transpose elements of the matrix are the inverse of the original values. The sum of the columns is divided into the elements of the column to give the normalized matrix. The principal vector is the mean of each row. In order to compute the consistency of the priorities, the weighted sum and weighted mean are computed. λ_{max} is computed from the weighted means and used to calculate the consistency index. From a table of random consistency [4], for n = 3, the value of 0.58 is obtained. The consistency ratio computed from the tabled value and the consistency index. The pairwise matrix, principal vector, weighted mean, λ_{max} and consistency ratio is given in table one. The consistency rule indicates that this value is on the border of being acceptable. The rule states that values greater than 0.10 *may not* be consistent. However, the observed value has a small deviation from the rule and the raw data indicate that the decision maker's preference is linear. Based upon these observations, the judgments can be considered to be consistent.

A pairwise comparison of the remaining decision criteria are constructed in a similar manner. Each criteria is compared to each other criteria and a square comparison matrix is formed. This matrix is termed a *priority matrix*. The principal vector of the matrix contains the relative importance of each criteria. The next step is to apply the pairwise comparison method to the criteria. Each criteria is compared to each other criteria and given a value. The principal vector is computed and can be interpreted as the ranked scores for the decision criteria. Table two presents the criteria scores after the criteria have been compared to each other. These scores indicate the relative importance of each criteria.

Table 1
Pairwise Comparison Matrix

	IBM PS/2	*IBM Compt.*	*MacIntosh*	*Principal Vector*	*Weighed Mean*
IBM PS/2	1.0	3.0	0.3	0.28	3.13
IBM Comp.	0.3	1.0	0.3	0.14	3.05
MacIntosh	3.0	3.0	1.0	0.57	3.23

λ_{max} = 3.14 Consistency Ratio = 0.118

Table 2
Decision Criteria Pairwise Comparison

Educational Criteria Row Mean		*Functional Criteria* Row Mean			
Homework	0.337	Space	0.238	Repair Cost	0.062
Student Preference	0.257	Personnel Availability	0.152	Vendor Market Status	0.037
Future Job	0.182	Software Cost	0.152	Peripherals	0.037
Compatible w/ Software	0.127	Compatible w/ network	0.152	Procurement Cost	0.096
Faculty Preference	0.095	Upgrade Cost	0.037	Vendor Reliability	0.037

In order to construct an overall comparison for the decision alternatives from among the criterion rankings and the priority matrix, a final matrix operation is required. The element of a decision alternative in the principal vector of the priority matrix is multiplied by the corresponding element in each of the decision criteria's principal vector. The resulting sum is the overall priority weight for each decision alternative. Table three presents the scores for the educational, functional and overall scores. The overall score was computed by summing the educational and functional criteria scores.

The results indicate that the MacIntosh personal computer is preferred choice when educational criteria are considered. In contrast, for the functional criteria, the IBM PS/2 has been assigned the highest score. By combining the educational and functional criteria scores, it is observed that the order of preference is: the IBM PS/2, the MacIntosh and the IBM compatible. This application of Analytic Hierarchy Process demonstrates a method for comparing three different computers over multiple criteria.

Integer Goal Programming Formulation

In order to determine "how many" of each type of personal computers to purchased, an integer goal program was developed. The Analytic Hierarchy Process produced a prioritized evaluation of each of the decision alternatives with respect to the perceptions of the decision makers. Each of the decision alternatives (IBM, IBM compatible and Apple MacIntosh) received an overall score. The overall score indicates the relative order of importance for each personal computer and assists the decision makers in determining where the Microcluster should place its resources.

Table 3
Analytic Hierarchy Process Rankings

	Overall Priority	*Educational Priority*	*Functional Priority*
IBM PS/2	0.770	0.391	0.378
IBM Compatible	0.539	0.209	0.330
MacIntosh	0.690	0.399	0.291

The priority rankings that were observed from the application of the Analytic Hierarchy Process can be incorporated into a integer goal programming model. Goal programming is a linear programming method that is appropriate to use when the purpose is to satisfy multiple conflicting objectives under the constraints of scarce resources [3]. Goal programming has the capabilities to incorporate the decision maker's preference system for multiple conflicting goals. In goal programming, all of the objectives are assigned target levels (the goals) for achievement and a relative priority is placed upon achieving these levels. Goal programming will find an optimal solution by allocating resources to these objectives in the order of the specified priorities. [3]

The decision variables for the model formulation are:

Integer Decision Variables

I1	the number of IBM PS/2 computers
I2	the number of IBM compatible computers
I3	the number of MacIntosh computers

Goal Priority Variables

P1	the overall AHP priority value for IBM PS/2 computers
P2	the overall AHP priority value for IBM compatible
P3	the overall AHP priority value for MacIntosh computers

Goal Deviational Variables

d1-	underachievement of the target number of IBM PS/2 computers
d1+	overachievement of the target number of IBM PS/2 computers
d2-	underachievement of the target number of IBM compatible computers
d2+	overachievement of the target number of IBM compatible computers
d3-	underachievement of the target number of MacIntosh computers
d3+	overachievement of the target number of MacIntosh computers

There are two type of constraints in this system. System constraints describe the limitations of physical space and financial resources. The goal deviational constraints model the shortfall and surplus of personal computers that will satisfy the AHP priorities. Data for these constraints were obtained by interviewing the director of the microcluster. It is important to note that the data are hypothetical. While no attempts were made to obtain actual data for the IGP, if such data were made available, it would a simple matter of substituting these data into the contraints.

System Constraints: The budget for this project was projected to be $35,000. The estimated price for each type of computer are: IBM PS/2 ($2,500), IBM compatible ($2,300) and MacIntosh ($2,700). This constraint is formulated as: $2500*I1 + 2300*I2 + 2700*I3 <= 35,000$ *(dollars)*. Each IBM PC requires 28 square feet of physical space. Each MacIntosh requires 23 square feet. The total amount of physical space available in microcluster was estimated at 500 square feet. This constraint is formulated as: $28*I1 + 28*I2 + 23*I3 <= 500$ *(square feet)*. Two thousand dollars was set aside for the purchase of computer peripherals. This amount includes the network connections and assorted utilities. The price per each IBM PS/2 and compatible is $100. The price for each MacIntosh is slightly higher, at $125. This constraint is formulated as: $100*I1 + 100*I2 + 125*I3 <= 2,000$ *(dollars)*. The sum of $3,000 dollars was allocated for the purchase of computer software. The price of software for each IBM PS/2 and compatible is $200. The price of software for each MacIntosh is slightly higher, at $250. This constraint is formulated as: $200*I1 + 200*I2 + 250*I3 <= 3,000$ *(dollars)*.

Goal Deviational Constraints: The last requirement is that the Microcluster acquire a representative cross section of computers so that students can have the opportunity to try all types of computers. These constraints are modelled as aspiration or goal constraints. The goal is to try to obtain at least seven of each type of computer and thus satisfy the requirement. Since the integer goal program may determine that it is optimal to purchase either a shortfall or surplus of a particular type of computer, the deviational constraints will allow the program to construct a feasible integer solution to the problem.

$I1 + d1- + d1+ >= 7$ *IBM PS/2 requirement*
$I2 + d2- + d2+ >= 7$ *IBM compatible requirement*
$I3 + d3- + d3+ >= 7$ *MacIntosh requirement*

Integer Goal Programming Objective: Minimize $P1*d1- + P2*d2- + P3*d3-$. Substituting the AHP priority values into the goal weights gives the following objective: $0.748*d1- + 0.471*d2- + 0.675*d3-$.

Integer Goal Programming Solution: The optimal solution obtained from the program is to buy seven IBM PS/2 computers, buy six MacIntosh computers and buy no IBM compatible computers. The solution is consistent with the priority weights obtained from the AHP. The integer goal program model satisfied the minimum requirement for the highest AHP priority weighted goal (IBM PS/2) and then attempted to satisfy the next highest AHP priority weighted goal (MacIntosh). Resource limitations prevented the model from completely fulfilling the MacIntosh minimum requirement and indicated that it would not be feasible to purchase IBM compatible.

Examination of dual integer linear program output indicates the following: The amount of money spent on computers was $33,700. The amount of money spent on peripherals was $1,450. The amount of money spent on software was $2,900. The amount of square feet that was used was 346. The total amount of money left over was $1,950.

The leftover amount is insufficient to purchase another computer. If funds were available, the Microcluster has the physical space to support additional computers. The binding dual variables were the number of IBM compatible and MacIntosh computers.

Conclusions

The management of the growth of a computer center is a complex problem. As university computer centers integrate personal computers into their array of computing facilities, the challenge of selecting the appropriate software and hardware becomes increasingly difficult. Since the software is dependent upon the hardware platform, the primary decision that computer centers face is to determine the hardware that will best support their educational goals and objectives.

One approach to answering the questions of *"what kind"* of personal computers to purchase and, given a limited budget, *"how many"* of each type of personal computers to purchase was presented in this paper. The approach employed two multiple criteria decision analysis methods to determine the answer to these questions. In order to determine *"what kind"* of computers to purchase, Thomas L. Saaty's Analytic Hierarchy Process was applied to determine the priority values for three types of personal computers. The priority rankings for each type of personal computer were determined from a hierarchy that was based upon fifteen criteria. Personal computers were compared on a pairwise basis on each of the fifteen criteria. Next, the criteria were compared on a pairwise basis. This produced an ranked score for each personal computer on each of the criteria and a ranked score for each of the criteria. Multiplying these scores provides a summary score for each of the personal computers.

Based upon the priority score information, an integer goal programming model was formulated. The integer goal programming model considered the cost constraints (personal computer hardware, network hardware and software) and physical space constraints. Additional constraints could easily be incorporated into the integer goal programming model (such as maintenance costs or user/computer ratios). The model was used to determine the optimal type and number of computers by satisfying goals given the constraints.

The advantages of applying Multiple Criteria Decision Analysis methods to complex problems are that these problems can be decomposed into smaller, solvable problems and later unified into a coherent solution. The Analytic Hierarchy Process provides a methodological process for the quantitative comparison of personal computers over a variety of criteria. The results of the Analytic Hierarchy Process are enhanced by linkage to the powerful optimization method of integer goal programming. The resulting combination of multiple decision analysis techniques provides a quantitative, objective process for selecting personal computers for a university personal computer cluster.

References

[1] Seidmann, Abraham and Ami Arbel, "Designing a Multimicro Distributed Accounting Information System", Human Systems Management, Elsevier Science Publishers B.V. (North-Holland), Volume 5, Issue 4, 1985. pages 283-291.

[2] Seidmann, Abraham and Ami Arbel, "Microcomputer Selection Process for Organizational Information Management", Information and Management, Elsevier Science Publishers B.V. (North-Holland), Volume 7, Issue 6, 1984. pages 317-329.

[3] Lee, Sang M. Introduction to Management Science, The Dryden Press, 1983.

[4] Saaty, Thomas L. The Analytic Hierarchy Process, MacGraw Hill, 1980.

Biography

Philipp A. Djang holds M.S. degrees in Educational Management and Development and Industrial Engineering from New Mexico State University. He is currently a doctoral student. He is employed as an Operations Research Analyst with the U.S. Army Training and Doctrine Analysis Command at White Sands Missile Range.

Special Presentation—ISTE

Network 2001: The Future of Telecomputing in Schools

Each panelist will present his/her vision of where telecomputing may be headed over the next ten years. Recent developments in technology, research results, trends in projects and efforts to develop the infrastructure will be discussed.

Moderator:

Chris Clark
Pennsylvania State University

Panelists:

Bob Spielvogel
Learning Link

Cecilia Lenk
TERC

Yvonne Andres
FrEdMail

Margaret Riel
AT&T

Jason Ohler
University of Alaska

Lynne Schrum
University of Oregon

Jim Levin
University of Illinois

Dan Lake
BOCES, New York

Projects
Multimedia and Science

The Weather Project: A Coast-to-Coast Telecommunications Project

Eadie Adamson, The New Laboratory for Teaching and Learning, The Dalton School, New York City
Carol Goodman, Campbell Hall, North Hollywood, California

Abstract

The Weather Project was begun in September 1990 in order to explore the use of *LogoExpress* for telecommunications between fourth grade students. The Dalton School in New York and Campbell Hall in North Hollywood began a very active exchange, joined later by a sixth grade class from Mann School in St. Paul, Minnesota. Although collecting temperature data and graphing the results with *LogoWriter* was the original basis for the exchange, we saw great potential for introducing students to telecommunications as a vehicle for sharing projects and ideas with other Logo classes. At Dalton students worked on weather-related projects. Study of the structure of snowflakes and observation of snowstorms resulted in several *LogoWriter* projects. One class worked on an extensive project about the three participating cities, using *LogoExpress* to request information from St. Paul and North Hollywood.These projects were sent via *LogoExpress* to the other two sites. Students from the receiving schools sent back letters with comments on the projects. Students at Campbell Hall extended their Logo knowledge as they learned to create graphs of high and low temperatures. During the period leading up to and during the war in the Persian Gulf, Campbell Hall students found a novel method of exchanging their thoughts on war. Dalton students received and read their communication and reciprocated with their responses. During the year, *LogoExpress* also served as an important vehicle for exchange between the teachers. We have begun to draw some conclusions about how telecommunications exchanges between groups of schools can be managed effectively, what kinds of projects might be reasonable to share, and how to preserve a sense of personal communication within the context of a group venture. Samples of graphs and projects will be shown.

Girls, Computers, and Science: Involving Them with Real World Problems

Diane McGrath
Kansas State University

Abstract

In a large suburban school district, middle school science teachers are working to enhance both the quality of their science teaching and equity in science through the use of cooperative learning strategies combined with excellent software, notably *Voyage of the Mimi*, *Second Voyage of the Mimi*, and *KidsNET*. The project, funded by an Eisenhower Mathematics and Science Education grant, involves training these 24 teachers in the use of computers and multimedia during the 1990-91 school year, and assessing the immediate impact of the equity strategies and materials on this year's students. The computer software has been chosen to fit in with the district's new FAST science curriculum. The district and the trainers have emphasized environmental science for all three years of middle school science. it is expected that this topic, combined with software/multimedia which provides motivation, curriculum integration, and a wide variety of role models, will enhance the interest and involvement of all children in science, and will particularly improve girls' attitudes. Evaluation of the project will be reported with a particular focus on the teachers' sense of computer efficacy, microcomputer concerns, and actual use of computers; and the attitude measures of students toward science and toward computers.

ENLIST Micros III: Models for Establishing ENLIST Micros Teacher Centers

James D. Ellis and Phillip G. Goulding
Biological Sciences Curriculum Study (BSCS)
Colorado Springs, Colorado

Abstract

The Biological Sciences Curriculum Study (BSCS), with support from the National Science Foundation (NSF), is conducting the third phase of ENLIST Micros. ENLIST Micros: Phase I developed a set of multimedia materials (text, instructor's guide, videotape programs, and computer software tutorial) that present the knowledge and skills that science teachers need to use microcomputers in science teaching. Phase II developed a staff development model (leaders and building teams attending workshops and seminars and supporting each other) for implementing microcomputers in science teaching. ENLIST Micros: Phase III, a 42-month project that began February 1989, builds on the first two phases to achieve the following major goals:

- Help teachers (K-12) improve their use of microcomputers in science teaching.
- Establish a national network of ENLIST Micros Teacher Centers.
- Determine the ability of regional sites to sustain a teacher enhancement project while phasing out external support.
- Disseminate information about how to establish successful, self-sustaining teacher centers for improving science teaching.

During 1989-91, BSCS established ten new teacher centers for ENLIST Micros. BSCS provided the staff from each of these centers with a two-week summer institute, two follow-up leadership seminars, two site visits, and consultative support via telephone and computer bulletin board. During 1991-92, BSCS will provide training to staff from four or more additional regional teacher centers and continue to provide follow-up support to the network of 15 or more centers.

Projects
Lego Logo Projects

A Jump Start for Future Scientists and Future Science Teachers

Kathleen Sullivan, Mathematics Department
Seattle University, Washington

Abstract

During the summer of 1990 Seattle University offered "Jump Start," computer-basketball camp for economically disadvantaged children between 10 and 12 years old, with Seattle University education majors serving as "coaches." The first week of the camp was for boys, the second for girls. Each team of youngsters spent mornings working together on the computer, and played basketball together in the afternoon.

Using the Lego-Logo materials, they built "robots" out of Lego parts and ran them using an interface to the apple IIGS. Since the materials were new to all of the campers, they started out from scratch together. The instructions were heavily visual and so a lack of reading skill was not a big problem. Most importantly, the children experienced the advantages of team work in academics as well as athletics. They found that just as simple commands can make sophisticated programs, a team of builders can accomplish much more than one working alone. Moreover, the future teachers had the gratifying experience of guiding the youngsters who were discovering mathematics and science and having a great time in the process.

Creating a Cooperative Learning Environment Using LEGO TC Logo

Jo Ann Karaffa
Spring Mill Elementary School
Indianapolis, Indiana

Abstract

In the last two years, I have become quite interested in utilizing strategies of cooperative learning in my fourth grade classroom. I have found that many of these strategies can be effectively used in conjunction with LEGO TC Logo activities.

I would like to share how I create a learning environment in my classroom so that otherscan gain from my experiences. I will present lesson plans and units of study that I have created along with the specific cooperative learning strategies that I use. These strategies are implemented with one computer in the classroom and just one Technic Control O Set of LEGO TC Logo.

Bridging the Gender Gap: LEGO-Logo Projects including SPACE STATION 2000

Manorama Talaiver
Mathematics ansd Science Center
Richmond, Virginia

Abstract

Qualitative study of LEGO-logo projects examines sex differences in designing, creating, operating, programming, and presenting inventions. The projects were programs demonstrated and taught at the Mathematics and Science Center and the Consortium schools. In addition students were observed during Space Station 2000, a Governor's Summer School Program at the Mathematics and Science Center. Space Station 2000 integrated physical science, space science, robotics, and Logo programming. Instructional period of Space Station 2000 was about 30 hours. The students were given instruction on the essentials of Space Station Freedom, Simple machines and in particular gearing system, movements with robotic arms, and introduction to primitive commands in LEGO-TC-Logo. Students telecommunicated with other educators on EDTECH via BITNET about their projects and received feedback. Each group chose a module and built a device or a machine to be placed in the module for their final project. In addition, science lessons were demonstrated at the school sites for an instructional period of 12-15 hours in five schools. A descriptive analysis of student behavioral outcomes will add suggestions for improving female and minority participation in science instruction.

Special Presentation

The Impact of Computers on the Art Curriculum: Three Case Examples

The impact of computers on the art curricula at three different grade levels is described in this invited presentation. Each presenter will describe the computer equipment, the graphics programs and the computer-based graphics projects that are employed in their general art curricula and in extra curricular activities. Ms. Kepler will describe an art option program for at risk children, an interdisciplinary art activity, and a general art lesson, all of which involve students in the use of computer graphics at the elementary level. Ms. Hololob will describe general art activities used to produce award winning computer graphics at the secondary level. Dr. Higgins will describe computer and photographic technologies used to produce graphic materials for projection in a presentation graphics course at the post-secondary level. Examples of students' computer graphics will be displayed

Panelists:

Loretta Kepler
Fifth Avenue Elementary School
East Northport, New York

Terri Holob
Locust Valley High School
Locust Valley, New York

Norman Higgins
Arizona State University
Tempe, Arizona

Projects
"Write On" Intermediate Writing

The Mystery Disc: Encouraging Writing in Junior High School

Connie Gonser, Rosemary Lyons, and Sylvia McGann
Royal Oak, Michigan

Abstract

English teachers are always trying to discover new ways to encourage students to write. As we become more and more a video culture, writing becomes less and lessenjoyable to students. Two teachers in Royal Oak, Michigan have developed a unique method for developing interest in writing on the part of their junior high school students. They capture their interest with a videodisc, The Mystery Disc. The Mystery Disc is one of the first produced in this country. It was never intended for instructional use, but rather was designed to be an interactive videodisc that replicated the game Clue. In Royal Oak The Mystery Disc becomes the resource that allows us to create a detective agency, an investigation with associated interviews, depositions, reports, etc. The students watch a "murder" (there are 16 on the disc), then interview the other students who are playing selected roles, write up their interviews, await reports from the detective agency, and participate in a large number of writing activities. This approach has been unusually successful in stimulating writing by junior high students. In this session we will share the HyperCard stacks we use that drive the selection of the specific murder, the support materials we use to encourage writing, and a sample of the writing done by the students.

Teaching and Learning with Technology

Janet Van Dam
Oakland Schools, California

Abstract

The Teaching and Learning with Technology Project was funded by Oakland Schools in 1987 to bring together in an elementary school those technologies that will be key components of schools in the future with an instructional program designed to prepare students for the Information Age. Spring Mills Elementary School in the Huron Valley School District was identified as the site for a pilot project that would enable personnel from both the local school district and Oakland Schools to accomplish the following three objectives:

- Explore and identify appropriate classroom applications for technologies such as microcomputers, instructional television, videodiscs, CD-ROM, and telecommunications.
- Design staff development programs that result in effective use of technology to accomplish curriculum objectives.
- Study what impact technology can have on teaching and learning.

The project focused on third, fourth, and fifth grade classrooms within Spring Mills Elementary School. In Fall,1987, one class at each of those levels was selected to serve as a Teaching and Learning with Technology (TLT) classroom. Students in those classrooms progressed in subsequent years to the TLT classroom at the next grade level.

In Spring,1990, the first group of students to complete three years in the TLT program graduated to middle school. The evaluation of the program was conducted with a primary focus on this first three-year TLT cohort, while including information obtained from students over the past three years who were in the program less than three years.

Using Hypermedia to Elicit the Stealth Essay

Gary Brown
San Diego State University

Abstract

At the Academic Skills Center at San Diego State University, we've been using NTERGAID'S Hyper Writer MS DOS based hypermedia authoring program to develop curriculumfor our developmental reading and writing students. We've put the Center's required readings into hypermedia, with graphics, vocabulary, supplemental readings, summaries, criticisms, schema, news clipings, grammar, etc. all linked to the main text. Students navigate through the essay, reading and frequently responding to various prompts. We've created paths in the text which elicit student writing in several rhetorical modes which areeventually concatenated along with pre-written transitions to create the entire "stealth" essays or rough drafts analyzing the curriculum reading.

Each of the paths focuses on a specific student deficiency. The writing elicited by the program provide a student generated model which helps students relate through their own writing to a reading which previously may have seen like academic abstraction. Concurrently, student writing is informed by the process of reading. Teachers, on the other hand, are freed from much deadening repetition and are provided with a tool which increases individualized instruction.

Our comprehensive departmental research maintained by a data Sbase has established surprising and significant gains in our computer based classes.

Special Presentation
Report from the ACM Two-Year College Curriculum Task Force

The Association for Computing Machinery (ACM) Two-Year College Computing Curriculum Task Force is actively working toward the generation of recommendations on curricula and related issues relevant to the two-year college computing environment. This includes identifying appropriate outcomes/goals of computing course groupings and curricula at two-year colleges as well as identifying content pertinent to the outcomes/goals for each of five selected areas. The areas under consideration are computer engineering and technology, computing in information processing, computing sciences, computer support services, and computing for other disciplines. The panelists who are members of the ACM Task Force Steering Committee, will report on the progress within each area.

Panelists:

Moderator:
Karl J. Klee
Chair, ACM Two-Year College Task Force
Jameston Community College
Jameston, NY

Co-Presentors:
Richard Austing
University of Maryland
College Park, MD

Helene Chlopan
University of Kentucky
Lexington, KY

John Impagliazzo
Hofstra University
Hempstead, NY

Joyce Currie Little
Towson State University
Baltimore, MD

Paper
Using the Scientific Method in Introductory Course Lab Exercises

R. Dale Johnson
Department of Computer and Information Systems
Jacksonville State University
Jacksonville, AL

Abstract

This paper describes the development of lab exercises for an introductory computer programming course. The exercises make use of the scientific method to allow students to "discover," through experimentation: (1) what causes common syntax errors and how to interpret compiler messages so that these errors can be corrected;(2) what causes common run-time errors and how to remove them; and (3) how much easier it is to repair a well-documented, modular program than to repair a short, undocumented one.

In most cases, the students see a language construct or run-time error in the lab before it has been discussed in class. Once they have completed the lab, the classroom discussion builds on what they discovered in their experiments. Even students who do not successfully complete the lab have a deeper understanding of course topics than they would have if they had not even attempted the lab.

Introduction

This paper describes the development and use of lab exercises for an introductory programming class. The lab exercises are based on the scientific method and require the student to discover through experimentation general principles about their compiler, their text editor, or their computer. In most cases, the student is learning something in the lab that has not been discussed in class. Usually it is after the lab exercise has been completed that the students receive a classroom explanation of what happened in the lab and why. It is hoped that the classroom discussion is then less abstract and more meaningful and the students will ask more questions. This situation is analogous to an auto mechanics course in which students drive a car in need of repair before they are told about whatever is broken on the car. It seems that the students would have a better appreciation of power steering, fuel injection, the importance of oil, and so on, after seeing how these affect the performance of the car.

The Scientific Method

The scientific method is "a formalization of learning by experience" (Smith, 1963). It involves three steps:

(1) observation of some natural process,
(2) formation of a hypothesis to explain the process, and
(3) experimentation to confirm/refute the hypothesis.

The three steps form a loop since further experimentation usually involves making more observations. The word "model" can substitute for the word "hypothesis" since one is often attempting to form a model of some part of nature.

The scientific method can be viewed as a combination of induction (going from the particular to the general) and deduction (going from the general to the particular). The hypothesis is formed by induction because particular observations are used to form a general rule. Deduction is involved when the hypothesis (a general rule) is used to predict future particular observations (Smith, 1963).

The scientific method is easily used in the natural sciences as a teaching tool. Using previous experiments as a guide, students can be asked to form a hypothesis that predicts what will happen in a new experiment. For example, Ford (1987) describes how high school students were asked to predict enzyme binding after a general discussion of enzymes. In this particular experiment, very few students correctly predict what will happen. However, even if a majority of the students form a poor hypothesis and seem to "fail" the lab, they learn something. Any classroom lectures that follow the lab are more meaningful to the student.

It seem obvious that any of the natural sciences would benefit from such a teaching approach. However, Bryant and Marek (1987) report that in our schools, 90 percent of science teachers spend 95 percent of their time using only a textbook. They ask why teachers always teach music and art with a "hands-on" approach, yet expect students to learn a science using only a book. They present a very positive overview of how satisfied both teachers and students are with an approach, based on the scientific method, that allows the students to become "responsible for their own learning."

If teaching and learning a natural science are best accomplished through a scientific method, self-teaching approach, why do so many computer science courses rely so heavily on textbook and chalkboard learning? Why not let computer science students "discover" how their text editor works, where semicolons are needed in a Pascal program, or how to delete a file through experimentation? The compiler, the operating system, and the editor can be treated as "natural" phenomena whose basic working principles the student is to discover.

The Scientific Method in the Computer Science Lab

The lab exercises described in this section are being used in an introductory programming course using Pascal. Many class meetings begin with a lab exercise, which generally takes fifteen to twenty minutes. Each lab has a form that the students fill out to describe what they tried, what they learned, and so on. Some lab exercises have an extra question at the end to entertain students who finish the lab early. The exercises are not currently graded in order for the students to know that their *attempts* at solutions are more important than outright success.

The lab exercises concentrate on exposing the students to the typical syntax and logic errors with which beginning students are confronted. There is a great deal of emphasis on fixing incorrect programs rather than on running a correct program and making enhancements to it. Most exercises therefore begin with the students attempting to execute a program and either seeing a syntax or execution error message. In many cases, the students have not been prepared by a classroom lecture for what they will see in the lab. Only after the lab are they told exactly what happened and why. What follows is a detailed discussion of three lab exercises.

Lab 1 (the worksheet of which is shown in Figure 1) was given on the first class meeting. The students had received

LAB #1

Purpose: Learn how to log onto the system and how to execute a program. Learn about INPUT-PROCESS-OUTPUT.

Problem: You are to execute the program in file LAB1 and determine what it does. All you know is that the program requires two inputs and produces one output. To execute the program, type

```
EX LAB1
```

and when you see a question mark, type in two integer values separated by a blank.

Observation:

Input 1	Input 2	Output
______	______	______
______	______	______
______	______	______
______	______	______

Hypothesis: Write here what function you believe this program performs.

1.

2.

3.

Experimentation: Purposefully try something that might refute your hypothesis. If you do, go back and state another hypothesis and attempt to disprove it.

Input 1	Input 2	Output
______	______	______
______	______	______
______	______	______
______	______	______

Figure 1

LAB #2

Purpose: Learn one of the most important and most confusing syntax rules of Pascal. Learn that a program cannot execute if it contains syntax errors. Learn how error messages are reported and how cryptic they can be.

Discussion: Many programming languages use semicolons as part of their basic syntax, but what role the semicolons play can differ from language to language. The two basic uses of semicolons are as terminators or separators. If they are used as terminators, a semicolon should appear at the end of every statement in a program. If they are used as separators, they are only used to separate one statement from a following statement. (That is, if a statement is not immediately followed by another statement no semicolon is needed.)

Problem: You are to execute the program in file LAB2. You'll notice that it has many errors and will not execute. It will not execute because all the semicolons are missing and you are to attempt to determine how semicolons are used in Pascal.

Observation: Place semicolons where you think they belong in the program. Report what happens.

Hypothesis: Based on your observations, state how you believe Pascal uses semicolons.

Experimentation: Purposefully try something that might refute your hypothesis. If you do, go back and state another hypothesis and attempt to disprove it.

Question: If you were able to redesign Pascal, would you prefer to use semicolons as separators or terminators?

Figure 2

absolutely no computer instruction (and several could not believe that they would be expected to use the computer on the first day). The program they executed computed and printed the average of two numbers. There was no input prompt and the output was not labeled. They could not look at the source code simply because they did not know how to use the editor. Their task was to determine through experimentation what function the program performed. The true purpose of lab 1 was to let the students get the "feel" of how the computer does things. The students learned how to log on, how to change their password, how to execute a program, and how to log off.

Of the 22 students participating, 16 correctly guessed what the lab1 program was doing. Many tried to input real values (and the program expected integers), causing the program to crash, but as a result the subsequent discussion of data types seemed to go very smoothly. When they were shown the lab 1 program on the chalkboard, they asked many more intelligent questions than classes in the past (that didn't have labs) concerning variable names, begin/ends, and how readln finds its data.

Lab 2 (see figure 2) presented the students with a Pascal program that contained no semicolons. They were to insert semicolons and have the program successfully execute. They had received no discussion of semicolons previously. Lab 2 let the students learn how to use their text editor to repair a program and let them see how cryptic error messages can be.

Of the 19 students participating, only 7 were able to repair the program. Most tried putting semicolons at the end of every line. The biggest problem they had with the lab sheet was succinctly describing what they had tried.

Lab 4 was the first lab asking them to track down a logic error. The absence of parentheses in a formula lead to an incorrect answer. They had received no instruction on the priority of operators. The purpose of this lab was to show them that a program can be incorrect yet execute successfully and to show them the importance of hand-checking results. (Students are often so glad to have a program finally execute that they often never notice severe logic errors.)

Of the 19 students participating, eleven corrected the program successfully. Several of those that failed simply changed

LAB #4

Purpose: Learn about mathematical operations in Pascal. Learn about formatted output. Learn that a program can be incorrect yet have no reported errors.

Problem: The program in file LAB4 was written by a person who doesn't know Pascal very well. They assumed the program would produce 75 as output but instead the program printed 43! What went wrong?

Observation: Attempt changing the program so that it does produce 75 as an answer instead of 43. You can *ADD* things to a statement but you may not *DELETE* or *REWRITE* a statement. Describe what you tried.

Hypothesis: Based on your observations, state what you believe the cause of the error is.

Experimentation: Purposefully try something to refute your hypothesis. If you do, go back and state another hypothesis and attempt to disprove it.

Question: What do the colon and 5 after ANSWER in the WRITELN statement do? Try removing them. Does the output look any different?

Figure 3

the values of variables to get 75 as an answer! (The instructions were not clear on what could and could not be tried.) The subsequent classroom discussion of operator precedence went very quickly.

Other labs emphasize such typical Pascal problems as

- assigning a real value to an integer variable,
- if statements with semicolons incorrectly placed,
- comparing two programs that perform the same task and determining which is easier to read and understand,
- an unclosed comment,
- how to print a program,
- why if/then/else is often better than a sequence of if/thens
- scope of variables, and
- typical function and procedure errors.

Conclusions

In nearly every instance, the lab exercises seemed to make the classroom discussions go more quickly. The labs showed the students how important even the most insignificant Pascal rule can be; they seemed more attentive and inquisitive in class as a result, wanted to completely understand each concept, and had more "what if you do this?" questions. Warnings from the instructor about things they should never do and things they must always do were more meaningful to the students. They had already seen the disasters that could result from being careless.

An unexpected result of the lab exercises was that many (but not all) students seemed to enjoy the puzzle-like challenges the labs presented. On occasions when there was no lab exercise, some were actually disappointed and complained.

The most important benefit of these exercises was that the students became more independent of the instructor. Beginning students often know how they want to solve a problem but they end up spending so much time coping with the operating system and trying to decode error messages that they have little time to refine their solutions. For the students using these lab exercises, typical syntax errors were not baffling. They could concentrate more on solving the problem and less on semicolons and begin/ends. They saw their logic errors sooner in the problem-solving process.

Future Research

In the fall of 1991, the first two computer science courses at Jacksonville State will have a scheduled, mandatory lab similar to that of biology or chemistry classes. Classroom teachers will conduct the labs for their classes. This situation will allow further experimentation with lab exercises. It is hoped that we will be able to conduct experiments to determine if the scientific method approach really is a better approach.

References

Bryant, R.J. & Marek, E.A (1987). They like lab-centered science. *The Science Teacher*, 54(8), 42-45.

Ford, L.K (1987). The potato experiment. *The Science Teacher*, 54(8), 44-45.

Koffman, E.B. (1985). *Problem solving and structured programming in Pascal, 2nd edition*. New York: Addison-Wesley.

Walker, M (1963). *The nature of scientific thought*. New York: Prentice-Hall.

Paper

Teaching Twos Complement Arithmetic in the Classroom

Billy Wood
Arizona State University

Abstract

A 4-bit computer model is used as a tool to teach the concepts of twos complement arithmetic. The finite size of any computer is emphasized by using a 4-bit model to perform computations of any precision desired. Rules of twos complement arithmetic/algebra are presented. The necessity for flag bit testing for carry/borrow and/or signed overflow is demonstrated. The need for multiple precision addition and subtraction in any size machine is emphasized. The teaching tools required are a chalkboard and/or overhead projector. The student prerequisites are a knowledge of first year algebra and 6th grade arithmetic.

Twos complement arithmetic is almost universally used to represent positive and negative numbers in a computer. However, its use requires understanding and careful programming to prevent overflow or carry/borrow from introducing errors in results. This paper presents a set of proven rules for twos complement arithmetic and a well tried straightforward method of teaching them. The rules and teaching method require only a chalkboard and do *not* require a computer.

A complete overview of the twos complement number system is easily mastered if example problems are limited to 4-bit numbers. The rules learned for 4-bit arithmetic/algebra can then be applied to a computer of any size. Since there are only 16 different 4-bit numbers, it is easy to generate example problems. All 16 possible 4-bit numbers when considered as "pure magnitudes" are shown in the left column of Table A. They range from 0 = 0000 lowest to 15 = 1111 at the highest.

Table A
The Complete Number Environment of a 4-bit Computer.

Magnitude only	Signed "y" axis
15=1111	+7=0111
14=1110	+6=0110
13=1101	+5=0101
12=1100	+4=0100
11=1011	+3=0011
10=1010	+2=0010
9=1001	+1=0001
8=1000	0=0000
7=0111	-1=1111
6=0110	-2=1110
5=0101	-3=1101
4=0100	-4=1100
3=0011	-5=1011
2=0010	-6=1010
1=0001	-7=1001
0=0000	-8=1000

The twos complement number system splits these 16 available numbers in half. All numbers with the most signifigant bit equal to one (magnitude 8 or greater) are considered negative. Those with the most signifigant bit zero (magnitude 7 or less) are considered positive. This definition makes signed algebraic operations possible using computer arithmetic. Rewriting the 16 numbers as shown in the second column as if along the signed "y" axis of algebra emphasizes the position of the positive and negative numbers with respect to each other. The two columns of numbers then represent the entire environment of a 4-bit computer. All computer arithmetic operations, no matter how complex, or how large the numbers, can be demonstrated using these 2 column representations of 16 4-bit numbers. If a number is to represent only a magnitude with no sign attached the first column is used and the rules of carry/borrow are applied to keep the arithmetic correct. If a signed algebraic result is required the signed column of numbers is used and the rules of signed overflow are applied to assure those operations correct.

If any number is to be represented which is larger than that possible with only 4-bits, it must be represented as two, three, or more 4-bit numbers used together (called "double" or "multiple" precision.) This machine performs arithmetic on 4- bit numbers only. Larger computers with more bits available are faced with this same limititation. For example, an 8-bit machine reaches it's limits for numbers above and below +127 to -128. However no more complex operation or greater precision is possible with larger machines; only the speed of solution increases when more bits are used in each arithmetic operation.

Inside the arithmetic circuits of the computer these two number representations (magnitude and sign) occur with no interaction. The circuits of the computer perform the correct arithmetic for any solution using the rules given below. The programmer provides the interpretation of results required in each case.

The rules of addition of two binary numbers reflect that there are 8 different combinations of 3 input binary numbers. (The three input binary numbers are the augend, the addend and a carry from the next lower place value.) Here in Table C are the eight different combinations and their resulting sum and carry out conditions:

Table B
The Rules of Binary Addition

augend	0	1	0	1	0	1	0	1
addend	0	0	1	1	0	0	1	1
carry in	0	0	0	0	1	1	1	1
sum	0	1	1	0	1	0	0	1
carry out	0	0	0	1	0	1	1	1

The arithmetic of this computer and the normal rules of addition taught in the primary grades are the same, except for the use of binary numbers. However, finite size makes a computer need a carry/borrow "bit" or "flag" when addition results give magnitudes exceeding the register size. When signed answers exceed the register size of the machine an overflow "bit" or "flag" is required to give correct results. With these two additional bits (carry and overflow) the smallest computer has the same arithmetic capability as the largest computer available. The rules may thus be taught using the 4-bit machine on a chalkboard and will apply to any machine size. The rules to use with these two new bits are as follows. If addition results in a carry from the most signifigant place value of the numbers, set a "one" in the carry bit, otherwise set a "zero" there. If the signs of the addend and augend are the same, and the sign of the result is different, set a "one" in the overflow bit, otherwise set a "zero" there. These rules are the same as taught in a first year algebra class. Below are sample problems in this 4-bit environment to demonstrate the addition rules. The three columns illustrate the signed decimal problem on the left, the binary (base 2) computer solution in the center (including the carry and overflow bit), and the same problem interpreted as a decimal magnitude problem on the right. The "plus" inside parentheses means this is an addition problem.

	signed decimal problem	4-bit computer solution	decimal magnitude problem
augend	+5	= 0101	= 5
addend (+)	+1	= 0001	= 1
sum	+6	= 0110	= 6

Carry flag/bit = 0
Overflow flag/bit = 0

Example 1. Example of Normal 4-bit Addition

This example illustrates the usual arithmetic results. Two 4-bit numbers are added and the result has no carry or overflow. Therefore the results are correct in sign and magnitude.

	signed decimal problem	4-bit computer solution	decimal magnitude problem
augend	+4	= 0100	= 4
addend (+)	+5	= 0101	= 5
sum	+9	not= 1001	= 9

Carry flag/bit = 0
Overflow flag/bit = 1

Example 2. Addition Example Illustrating Signed Overflow

This example illustrates that the sum can "overflow" beyond the capabilities of the 4-bit number system to represent the signed result correctly, with no carry produced. Therefore the resulting magnitude is correct, but not the signed answer.

	signed decimal problem	4-bit computer solution	decimal magnitude problem
augend	-3	= 1101	= 13
addend (+)	-1	= 1111	= 15
sum	-4	= 1100	not= 28

Carry flag/bit = 1
Overflow flag/bit = 0

Example 3. Example Illustrating Carry Out With no Overflow

One in the carry flag determines that the magnitude result is incorrect (since it should be 28), but zero overflow indicates that the signed result is correct.

	signed decimal problem	4-bit computer solution	decimal magnitude problem
augend	-7	= 1001	= 9
addend (+)	-3	= 1101	=12
sum	-10	not= 0110	not=21

Carry flag/bit = 1
Overflow flag/bit = 1

Example 4. Addition Example Showing Both Carry and Overflow

Here both carry and overflow flags are "set" or "one" indicating that magnitude and sign are both outside the range of a 4-bit result.

These samples illustrate that 4-bit computer arithmetic occurs without interaction between sign and magnitude in the same problem. If the carry bit in each case is given a decimal value of 16 (the next higher place value) the magnitude results will all be correct. The sign overflow results are not as easily corrected. In each case the overflow bit did signal that the signed answer was correct when it was zero and incorrect when it was one.

To correct the signs of these examples the sums must be computed using the 4-bit machine in "double precision mode." As previously mentioned, each number is then represented using two 4-bit numbers taken together. Any carry from the least signifigant part of the sum must be added to the next higher place value (into the most signifigant part of the double precision number.) If double precision is large enough to encompass all the problems to be done, and the rules of double precision arithmetic are followed, the twos complement answers will be correct.

The remaining rules for double precision arithmetic are as follows. To represent a sign correctly in double precision arithmetic, the sign of the least signifigant part must be "extended" or copied into the most signifigant part of the problem. The sign of the least signifigant part of the solution is disregarded. The extended signs of Examples 2 and 4, above, are now corrected.

Table C
Double Precision Addition Examples

Double Precision Version of Example 2	Double Precision Version of Example 4
augend +4 = 0000 0100	-7 = 1111 1001
addend +5 = 0000 0101	-3 = 1111 1101
sum +9 = 0000 1001	-10 = 1111 0110

These results show that the precision of the numbers used in computer problems must be carefully chosen to prevent errors when using twos complement arithmetic. These same 4-bit examples may be used to teach subtraction.

Computer subtraction usually uses the following approach. Convert the problem to addition and then compute the result. Converting to addition is done by taking the "ones" complement of the subtrahend and adding one. (This produces the twos complement of the subtrahend.) The results will then be correct. An additional rule involves the carry bit (which must now be interpreted as a borrow.) It is the "inverted or not borrow" and must be complemented before placing into the carry/borrow flag bit.

Here are 4-bit examples to illustrate the subtraction rules:

The minus inside parenthesis indicates subtraction.

	decimal subtraction problem	4-bit computer problem	converted addition solution
minuend	+5	= 0101	= 0101
subtrahend (-)	+3	= 0011	= 1100
	added "one"		= 0001
difference	+2		= 0010

Carry/Borrow bit = 0 (since the actual carry was 1)
Overflow bit = 0

Example 5. Example of Normal Computer Subtraction

This result is like the first addition example. All the numbers in the problem "fit" inside a 4-bit system, including the difference. Thus the carry/borrow is zero and also the overflow.

(The minus inside parenthesis again illustrates subtraction.)

	decimal subtraction problem	4-bit computer problem	converted addition solution
minuend	-7	= 1001	= 1001
subtrahend (-)	-2	= 1110	= 0001
	added one	= 0001	
difference	-5	= 1011	

Carry/borrow bit = 1 (since the actual carry is 0)
Overflow bit = 0

Example 6. Subtraction Illustrating a Borrow and no Overflow

This problem illustrates the case where the magnitude part of the problem is incorrect, since the magnitude of -2 is greater than the magnitude of -7. The overflow bit does not set in this case since the signed result is correct for a 4-bit number.

	decimal subtraction problem	4-bit computer problem	converted addition solution
minuend	-3	= 1101	= 1101
subtrahend (-)	+6	= 0110	= 1001
	added one	= 0001	
difference	-9		not= 0111

Carry/borrow bit = 0 (since the actual carry is 1)
Overflow bit = 1

Example 7. Illustrates Subtraction Overflow with no Borrow

Here the carry/borrow bit does not set, since the magnitude of +6 is less than the magnitude of -3. The overflow bit is set indicating that the sign of the result is incorrect. In the next example both sign and magnitude are incorrect which cause the carry/borrow bit and overflow both to set.

	decimal subtraction problem	4-bit computer problemn	converted addition solution
minuend	-5	= 1011	= 1011
subtrahend (-)	+4	= 0100	= 1011
	added one		= 0001
difference	-1		not= 0111

Carry/borrow bit = 1 (since the actual carry is 0)
Overflow bit = 1

Example 8. Example Illustrating Overflow and Borrow

This example illustrates a subtle point about the magnitude of the two operands. The magnitude of +4 negated to produce -4 is greater than the magnitude of -5. (Note that the ones complement of 4 is identical to -5.) The additional one added to complete the twos complement causes the magnitude of the twos complement of 4 to be greater than the magnitude of -5. Also the sign of the result has overflowed since the sum of two negative numbers in the converted addition problem produced a positive result.

The two example subtraction problems with incorrect signs may be corrected using double precision arithmetic exactly as for addition. Here are Examples 6 and 8 redone using double precision. (Note the extended signs.)

original problem	converted to double precision addition
-7 = 1111 1001	1111 1001
(-) -2 = 1111 1110	0000 0001
additional one	0000 0001
difference result - 5 =	1111 1011

Example 6 Recomputed. Using Double Precision Subtraction

original problem	converted to double precision addition
-5 = 1111 1011	1111 1011
(-) +4 = 0000 0100	1111 1011
additional one	0000 0001
difference result -9 =	1111 0111

Example 8 Recomputed. Using Double Precision Subtraction

This paper has demonstrated the rules of twos complement arithmetic. Computer operation can be taught in a classroom by following these rules using only 4-bit numbers and a chalkboard. The student can master these new (to him/her) ideas using only concepts from first year algebra. Once learned, the rules apply to any machine of any size.

Special Presentation—HyperSIG
Hypermedia in Colleges and Universities: What are We Teaching and How Are We Teaching It?

Abstract unavailable at press time.

Paper
Achievement in a Computer-Rich Environment

W. George Cathcart
University of Alberta

Abstract

The achievement of two classes of grade three children in mathematics and language arts was monitored over a two-year period. The control class had virtually no access to computers. In the experimental class, each child had a computer on his/her desk with instant access for a half-day every day during the second year. During the first year there was one computer for every two children. The experimental classroom was an Apple Centre for Innovation partially funded by the Apple Canada Education Foundation. Special computer-based curriculum materials were prepared for language arts by the classroom teacher. In mathematics, commercial courseware was the primary source of computer-based experiences.

At the end of the first year the experimental class gained over five points on a 25-point writing scale compared to a slight decline for the control group. They also gained an average of 2.5 times as much as the control class in reading comprehension. On mathematical concepts the experimental class gained about twice as much as the control group but both groups gained about 12 points on the basic facts test. A statistical comparison of the groups on posttest scores after adjusting for pretest differences (one-way analysis of covariance) revealed a significant difference in favor of the experimental group on all language arts measures and on mathematical concepts.

During year-2 the experimental group made larger gains in writing than the control class and were significantly higher on adjusted posttest scores. However, on the reading test, there was no significant difference. In mathematics the year-2 analysis revealed that the experimental class had a significantly higher adjusted posttest mean on both the concepts and basic facts tests.

These results lend some support to the thesis that the computer used as a supplement to regular classroom teaching in a setting where children have a high level of access to the computer effects greater achievement than the absence of computer experience.

Recent surveys (Petruk, 1985; Hubert, 1988) have shown an exponential increase in the number of computers in schools during the past decade. Hubert's (1988) estimate translates into about a 1:15 computer to student ratio or about 100 minutes per week of computer access for each child in Alberta. The actual time a child spends at a computer, of course, varies significantly from this theoretical average. Questions remain. If children had continuous access to their own computer every day of the school year, what could they do? What would they learn? Would their thinking patterns change? Would their achievement level improve beyond what would normally be expected? How would the school program change?

Background
Setting

In an attempt to at least partially answer the broad and open questions stated above, an Apple Centre for Innovation (ACI) was established in a third grade classroom. The Centre was jointly funded by the Apple Canada Education Foundation and by Malmo School. The plan called for the installation of one complete Apple II GS microcomputer workstation for each child in the classroom and the development of materials that would uniquely integrate the computer into the language arts and mathematics programs.

The project began in the fall of 1987 and has expanded each year. This paper reports results from year-1 (1987-88) and year-2 (1988-89).

During year-1 the computer to student ratio was 1:2. In year-2 there was one computer for every student. A file server was added near the end of year-2.

Special furniture was designed to provide space for groups of four children to work together cooperatively or independently at their computer as illustrated in figure 1.

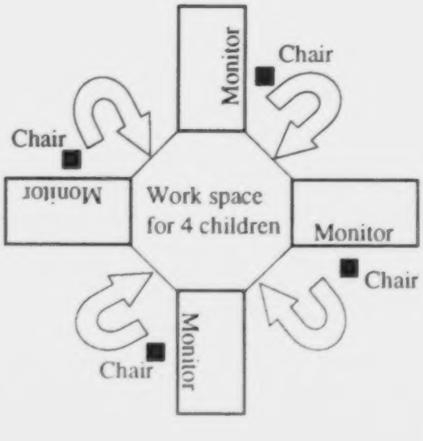


Figure 1. Octagonal Four-Student Workstation.

Research

Most components of the language arts program have felt the impact of the computer. Meta-analysis by Roblyer and King (1983) and Glass (1982) suggest that computer assisted instruction has a positive effect on achievement in reading, spelling, and writing.

The word processor has become an important writing tool in classrooms at almost all grade levels. Daiute (1982, 1986) claimed that computer writers are less concerned about making mistakes, compose longer manuscripts, and revise more. Fitzgerald (1987) argues that a greater amount of revision using a word processor is not a consistent research finding. More consistent are findings that suggest students using a word processor make more surface revisions whereas more expanded revisions occur with a pen.

Some studies (Kopp, 1985; Owston, Murphy, & Wideman, 1990) suggest that the same students, experienced with word processing, produced better quality work when using a word processor than when writing off-computer. Again Fitzgerald's (1987) review of research listed studies on both sides of this issue.

Levin, Riel, Rowe, and Boruta (1985) concluded that the "vast majority of the elementary school students we worked with preferred using a word processor, and their most common reasons were the ease of text input and editing. The power of a text editor is that it makes revision a real option for beginning writers" (p. 170).

Many of the earlier studies which investigated the effects of computer assisted instruction (CAI) on achievement reported no significant difference between CAI and non-CAI treatments. For example, Edwards, Norton, Taylor, Weiss, and Dusseldorp (1975) found about 44% of the studies they reviewed reported equal or mixed results when the achievement of CAI groups was compared to the achievement of groups taught in a "traditional" fashion. However, two theses significant for this study emerge from these earlier studies. First, elementary students seem to gain more from CAI than secondary or college students, and, secondly, CAI used as a supplement seems to produce greater gains than CAI used as a substitute for teacher instruction, especially at the elementary level. Conclusions similar to the above have been reported by Forman (1982), Hallworth and Brebner (1980), Roblyer (1985), and others.

More recent meta-analysis suggest a more positive relationship between CAI and achievement. Burns and Bozeman (1981) used six criteria to select 40 studies for a meta-analysis of the effect of CAI on mathematics achievement. They found an average effect size of 0.45 for tutorial CAI and 0.34 for drill and practice CAI. While recognizing the possible effect of many variables they conclude that "the analysis and synthesis of many studies do point to a significant enhancement of learning in instructional environments supplemented by CAI, at least in one curricular area—mathematics" (p. 37).

The findings from an evaluation of the Computer Curriculum Corporation's materials in the Los Angeles Unified School District were subjected to a meta-analysis (Glass, 1982). This project involved language arts and mathematics materials in grades three to six. CAI was used both as a supplement to and as a substitute for regular classroom instruction. Glass concluded that CAI is more effective in both mathematics and language arts when used as a supplement rather than a substitute for regular classroom instruction. The data seem to suggest that achievement gains are greater in mathematics than either language or reading.

Most of the studies reported in the literature were of rather short duration and computer access was often somewhat limited. What could young children do if they had virtually unlimited access to a computer with appropriate teacher intervention?

Purpose

School boards, governments, parent associations, and a variety of business and other organizations have spent millions of dollars putting computer equipment into schools. These agencies are interested in knowing what returns they are getting on their investment. Returns can be examined from many different perspectives. However, the one type of outcome supporting agencies often seem most interested in is achievement in the traditional school subjects.

The ACI setting described above where students had almost instant and unlimited access to a computer and where a determined effort was made to integrate the computer into the curriculum could be a fruitful setting in which to monitor achievement of the children.

The purpose of this report, then, is to

- document gains in language arts and mathematics made by students in a computer-rich environment over a two-year period, and to
- compare these gains with gains in a control class where the computer was seldom used.

Procedure

Selection of classes

The grade three class in the ACI served as the experimental group. A third grade class in another school with very limited computer access was selected as a control class. The experimental school was situated in a middle class urban district. The control school was situated in a slightly lower socioeconomic area in a different part of the city.

The control school had a well-established Arabic-English bilingual program. The experimental school had just started a similar program. Neither of the classes had students involved in this program during the first two years.

The control school was selected on the advice of central office personnel because its mean IQ score at the third grade level had been about the same as the experimental school's mean based on previous city-wide testing. The first year data revealed no significant differences in mean IQ or age scores between the two classes. The second year data indicated a slight IQ difference in favor of the experimental group.

Since intact classes were involved over the period of a full school year, the teacher factor was important. The only control on this was the judgment of central office personnel that all teachers involved were "good" teachers.

Treatment

In the fall a two-hour-per-week teacher directed keyboarding instructional program was provided for the experimental class. From January to June the keyboarding instruction time dropped off and became more of a maintenance program.

In the early part of the year, computer based language arts activities which required minimal keyboarding expertise were developed. Some of these activities made use of the word processor in *AppleWorks* (Claris, 1989) and considerable use was made of the database because less keyboarding was required. Once the children's keyboarding improved extensive use was made of the word processor. A significant part of the language arts program involved novel studies and theme studies using the word processor and database in *AppleWorks*. Reading comprehension questions and writing activities on each chapter of novels such as Beverly Cleary's (1984) *Ramona Forever* were put on a data disk. Each child loaded the appropriate file and responded to the questions directly on the computer. Journal, creative story, and poetry writing were also done directly on the computer.

In mathematics the computer was used primarily as a means of providing reinforcement. Courseware included *Mathematics Activities Courseware 3* (Super, 1984), a variety of products from the Minnesota Educational Computing Consortium, graphing activities from National Geographic's (1987) *Project Zoo*, some practice and problem solving software written by the project director, and an assortment of other material. The library of software available to children continually expanded during the two years.

Instruments

Tests of achievement in creative writing and reading comprehension were adapted from an earlier released version of a test administered on a province-wide basis (Alberta Education, 1984). For the creative writing portion of the test the children were given a booklet with an outline picture of two children at a beach. A brief story starter related to the picture was read to the children and they were then told to use their imagination to write the best story they could about the children in the picture.

The reading comprehension portion of the test consisted of four passages to be read; three prose and one poetry. Each passage had five multiple choice comprehension questions and one open-ended question associated with it. The passages and multiple choice questions were the same as those used by Alberta Education (1984). The researcher added the open-ended question for each passage.

The mathematics test was developed by the researcher based on the provincial curriculum and authorized textbooks. The test was designed in two parts; concepts and basic facts. The concepts portion consisted of open-ended and multiple choice questions. The concepts covered included number, operations, and properties, numeration, graphing, measurement, and geometry.

In designing the concepts test, a grid, content by cognitive level (knowledge, comprehension, application/problem solving), was used to ensure adequate coverage. Materials from the grade three authorized texts and from released versions of the achievement tests administered by the provincial government were used as the basis for the test items. The basic facts test included 12 items for each operation in multiple choice format. The test was designed to be at about a mid- to late grade three level in order to serve as a posttest as well as a pretest.

The Kuder-Richardson reliability for the concepts portion of the test using pretest scores was 0.78. Using the posttest scores the reliability was 0.81. The basic facts test was timed and therefore not included in the calculation of the reliability.

All tests were administered by the researcher over a two-week period from mid- to late September and again in late may to early June each year. A graduate student in language arts scored the writing and reading tests for each of the four administrations. A graduate student in mathematics scored the pre- and posttest in mathematics the first year. The researcher scored the mathematics tests in the second year. The mathematics test was objective so no discrepancy between scorers should have resulted.

Results

Each year the data was analyzed using a one-way analysis of variance with repeated measures to test the significance of gains and a one-way analysis of covariance to compare the posttest performance of the two groups after adjustments for pretest differences. After the second year, the data was examined with several multivariate analyses. Results of these analyses are reported below, but first a note about keyboarding.

Keyboarding

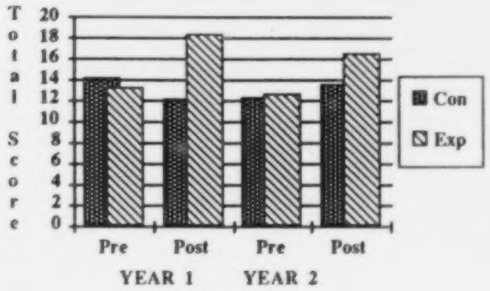
The children readily acquired proficiency in keyboarding. Results of the keyboarding instruction have been reported by Ubelacker (1988). After about 22 hours of instruction (early December) all students were keyboarding as fast or faster than they could write with pencil. By June over half of the third grade students were keyboarding 24 words per minute or better with the fastest student at 40 words per minute.

Year One

Only subjects with complete data on all variables were included in the analyses. This reduced the sample to 22 in each class. (Four students were eliminated from the experimental class because they were in grade two.)

Language Arts. The means for each class for both years are included in table 1. Figure 2 graphically shows the performance of each group over the two years on the total writing and reading scores. On the pretest the experimental class had the lowest mean on 7 of the 8 measures but on the posttest it had the highest mean on all 8 scales.

Group Comparison on Writing



Group Comparison on Reading

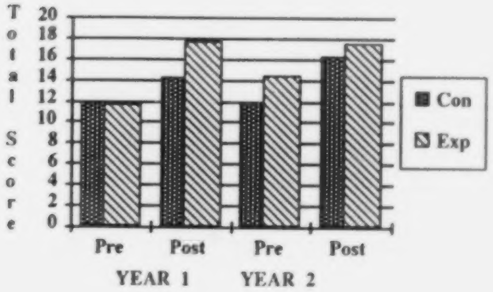


Figure 2. Comparison of groups on writing and reading over two years

Gains. During year-1 the control class declined on each of the five components of the writing task. The declines on the development and conventions subscales as well as on the total writing score were statistically significant. The experimental class made relatively large and statistically significant gains on each of the five components of the writing scale.

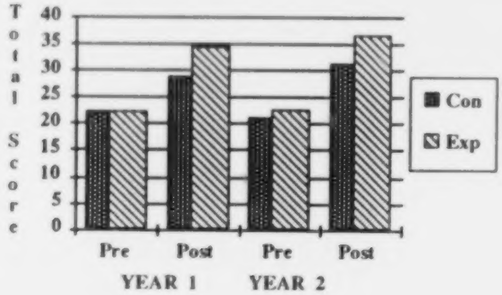
Both classes made positive and significant gains on the reading comprehension test. The experimental class gained over 2.5 times as much as the control class.

Class Comparison. After adjustments were made for differences in pretest scores, there were statistically significant posttest differences between the experimental and control classes on each language arts measure. Significant differences are indicated by bullets in table 1. On each of the eight measures, the experimental class scored significantly higher than the control class. On the total writing score the adjusted mean for the experimental class was over twice that of the control group. On the reading comprehension scale the adjusted mean of the experimental class was a little less than twice the mean of the control class.

Mathematics

The means for each class for both years were listed in table 1. Figure 3 compares the two groups on the totals of the concepts and basic facts sections of the mathematics test.

Group Comparison on Concepts



Group Comparison on Basic Facts

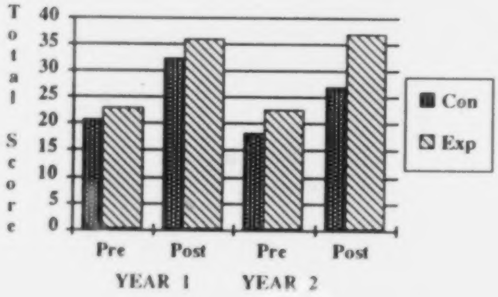


Figure 3. Two-year comparison of groups on math concepts and basic facts

Gains. It was expected that a significant positive gain in mathematics score would be made over the course of one school year. A one-way analysis of variance with repeated measures for each class confirmed, in general, this hypothesis but there were some exceptions.

The gains made by the experimental class were all statistically significant. The control class, however, had a total of three non-significant gains, all in the concepts portion of the test (graphing, measurement, and geometry).

Table 1: Pretest, Posttest, and Adjusted Means for each Group for each Year (Including indication of significant differences (•) between classes based on analysis of covariance)

			Year 1							Year 2				
	Pretest		Posttest		Adjusted			Pretest		Posttest		Adjusted		
Measure	Exp	Con	Exp	Con	Exp	Con	Sig	Exp	Con	Exp	Con	Exp	Con	Sig
Language arts														
Content	2.86	2.82	3.91	2.50	2.75	1.36	•	2.56	2.43	3.41	2.81	2.67	2.11	•
Development	2.91	2.95	3.86	2.18	3.13	1.43	•	2.53	2.43	3.41	2.33	2.54	1.50	•
Sentence Structure	2.45	2.82	3.55	2.50	2.04	0.77	•	2.44	2.19	3.06	2.62	1.40	1.13	
Vocabulary	2.55	2.77	3.64	2.68	2.50	1.45	•	2.66	2.76	3.34	2.95	2.86	2.45	•
Conventions	2.50	2.82	3.41	2.27	2.33	1.06	•	2.41	2.52	3.31	2.81	1.99	1.42	•
Total writing score	13.27	14.18	18.36	12.14	12.35	5.66	•	12.59	12.33	16.53	13.52	9.94	7.07	•
Reading comprehension	11.55	11.91	17.64	14.32	8.47	4.86	•	14.38	11.86	17.50	16.24	8.69	8.97	
Total language arts score	24.82	26.09	36.00	26.45	18.63	8.20	•	26.97	24.19	34.03	29.76	16.58	14.11	•
Mathematics														
Number, operations, and properties	9.55	11.09	18.14	15.73	11.83	8.40	•	10.63	9.57	18.69	15.52	14.39	11.66	•
Numeration	7.23	5.91	8.45	6.95	4.21	3.49		6.19	6.48	8.78	8.38	5.78	5.24	
Graphing	0.32	0.64	0.95	0.77	0.86	0.59		0.66	0.48	1.22	1.10	0.95	0.90	
Measurement	1.95	2.36	3.64	2.59	2.16	1.27	•	2.22	2.19	3.97	2.76	3.11	1.91	•
Geometry	2.64	2.36	3.64	2.59	2.16	1.27	•	2.78	2.38	3.84	3.43	2.95	2.67	
Problem solving	3.23	3.05	4.73	3.59	2.69	1.66	•	3.16	2.71	4.81	3.86	3.19	2.47	
Total concepts	22.14	22.36	34.32	28.68	18.76	12.96	•	22.47	21.10	36.50	31.19	22.81	18.34	•
Addition facts	10.23	10.00	11.86	11.59	10.75	10.50		9.47	9.19	10.94	10.14	6.48	5.82	
Subtraction	8.14	5.91	10.68	10.14	8.09	8.25		7.97	6.24	10.28	8.38	6.20	5.19	•
Multiplication	3.09	2.91	7.00	5.14	4.59	2.87	•	3.38	1.67	8.34	5.33	6.93	4.64	•
Division	1.59	1.77	6.36	5.41	5.00	3.89		1.94	1.14	7.44	3.14	6.21	2.42	•
Total facts	23.05	20.59	35.91	32.09	17.46	15.61		22.75	18.24	37.00	27.00	19.66	18.10	•
Total mathematics score	45.18	42.95	70.23	60.77	35.88	28.12	•	45.22	39.33	73.50	58.19	40.59	29.56	•

Class Comparison. After adjustments for differences in pretest scores, there were significant treatment effects on seven of the 13 scales (number, operations, and properties, measurement, geometry, problem solving, the total of the concepts test, multiplication basic facts, and the total mathematics score). On each of these scales the experimental group outperformed the control class.

Year Two

Complete data was available on 21 students from the control class and 32 from the experimental class. All means for year-2 were included in table 1 and means on the major scales were shown graphically in figures 2 and 3.

Language Arts

Gains. The experimental class had statistically significant gains on all language arts measures. The control class made significant gains on sentence structure and the reading comprehension test. The pre- to posttest change by the control group was not significant on the content, development, vocabulary, and conventions scales of the writing test. On the development scale there was a slight decline.

Class Comparison. The analysis of covariance found significant differences between the two classes on the content, development, vocabulary, and conventions subscales of the writing test. In all cases the adjusted mean for the experimental class was higher than the mean for the control group. On sentence structure there was no significant difference between the groups.

On the total writing score and the total of all language arts measures, the experimental class also had a significantly higher mean. On reading comprehension there was no difference between the two groups.

Mathematics

Gains. As in year-1, gains were achieved on all measures by both groups. The gains made by the experimental class were all significant. For the control class, all gains were significant except on the test of addition facts.

Class Comparison. On two of the six concept measures (number, operations, and properties and measurement), there were significant effects due to group. In both cases the experimental class scored significantly higher than the control class.

Treatment effects were significant on all the basic facts tests except addition. In each case the experimental group outperformed the control class. The experimental group also had a significantly higher mean than the control group on the total of the concepts test and on the total mathematics score.

Multivariate Analyses

After the second year, the data were subjected to four multivariate analyses. Each analysis included groups (two), year (two), time (two), and scores on a set of related scales. Two multivariate analyses were done on the language arts measures.

Table 2
Summary of Multivariate Analysis of Variance
Bullets indicate significant effects

Measure	Effect G*xYxT	Y x T	G x T	G x Y	Time	Year	Group
Language arts							
Content		•	•		•		•
Development		•	•		•		•
Sentence Structure	•		•		•		•
Vocabulary			•		•		
Conventions	•	•			•		
Total writing	•		•		•		•
Reading comprehension	•				•		
Mathematics							
Number, operations, properties			•		•		
Numeration		•			•		
Graphing					•		
Measurement			•		•		
Geometry					•		•
Problem Solving			•		•		•
Addition facts					•	•	
Subtraction facts		•			•		•
Multiplication facts		•	•		•		•
Division facts			•	•	•		•

* G= Group Y=Year T=Time (pre-post)

The first included the five scales on the writing test and the second included the total writing and reading scores. Two similar analyses were completed using mathematics scores. The first included the six subscores from the concepts portion of the test and the second included scores on the four basic facts test. Table 2 summarizes the results. The bullets indicate significant effects.

Summary and Conclusions

The results from this two-year study lend some support to the thesis that extensive use of the computer does have a positive effect on achievement in language arts and mathematics. In language arts the effect seems to be more apparent in writing than in reading. In mathematics the effects may not be as pronounced, but seem present in both understanding concepts and in knowledge of basic facts. Observing greater gains in language arts than in mathematics was opposite to the findings of Glass (1982). This might be explained by a greater emphasis on language arts in the experimental program than on mathematics.

The teacher factor was only minimally controlled in this study. This places some limitations on any interpretation of the results. Of course, over the span of a school year many other uncontrollable factors can be at play. However, daily and continuous access to the computer, along with specially prepared computer based curriculum was the outstanding factor that differentiated the groups. This, along with the magnitude of the differences in favor of the experimental group, suggests that the computer has to be considered as one factor that effects positive gains in achievement in grade three language arts and mathematics.

References

Alberta Education (1984). Student achievement testing program: Grade 3 language arts (English). *Student Evaluation Branch Bulletin*, 4 (10), Edmonton, AB: Alberta Education.

Burns, P. K., & Bozeman, W. C. (1981). Computer-assisted instruction and mathematics achievement: Is there a relationship? *Educational Technology*, 21(10), 32-39.

Claris Corporation (1989). *AppleWorks*. Mountain View, CA: Claris Corporation.

Cleary, B. (1984). *Ramona forever*. New York, NY: Dell Publishing Co.

Daiute, C. (1982). Word processing: Can it make even good writers better? *Electronic Learning*, 1 (4), 29-31.

Daiute, C. (1986). Physical and cognitive factors in revising: Insights from studies with computers. *Research in the Teaching of English*, 20 (2), 141-159.

Edwards, J., Norton, S., Taylor, S., Weiss, M., & Dusseldorp, R. (1975). How effective is CAI: A review of the research. *Educational Leadership*, 33(2), 147-153.

Fitzgerald, J. (1987). Research on revision in writing. *Review of Educational Research*, 57 (4), 481-506.

Forman, D. (1982). Search of the literature. *The Computing Teacher*, 9(5), 37-51.

Glass, G. V. (1982). Assessment of the effectiveness of computer-assisted instruction in the ETS-Los Angeles study. In M. Ragosta, P. W. Holland, & D. P. Jamison (Eds.), *Computer-assisted instruction and compensatory education: The ETS/LAUSD study* (Final Report #19). Princeton, NJ: Educational Testing Service.

Hallworth, H. J., & Brebner, A. (1980). *Computer assisted instruction in schools: Achievements, present developments and projections for the future*. Calgary, AB: Faculty of Education Computer Applications Unit.

Hubert, N. (1988). The 1988 Computers in Education Survey. *Computers in Education* 5 (7), 13-15, 17.

Kopp, R. J. (1985). *The effects of a word processor on the revision strategies of grade five writers when composing in the narrative mode*. Unpublished master's thesis, University of Alberta, Edmonton, AB.

Levin, J. A., Riel, M. M., Rowe, R. D., & Boruta, M. J. (1985). Muktuk meets Jacuzzi: Computer networks and elementary school writers. In S. W. Freedman (Ed.), *The acquisition of written language: Response and revision* (pp. 160-171). Norwood, NJ: Ablex

National Geographic Society (1987). *Project zoo*. Washington, DC: National Geographic Society.

Owston, R. D., Murphy, S., & Wideman, H. H. (1990). *On and Off Computer Writing of Eighth Grade Students Experienced in Word Processing*. Technical Report 90-1. North York, ON: York University, Centre for the Study of Computers in Education.

Petruk, M. W. (1985). *Microcomputers in Alberta Schools—1985*. Edmonton, AB: Alberta Education, Media and Technology Branch.

Roblyer, M. D. (1985). *Measuring the impact of computers in instruction: A non-technical review of research for educators*. Washington, DC: Association for Educational Data Systems.

Roblyer, M. D., & King, F. J. (1983). *Reasonable expectations for computer-based instruction in basic reading skills*. Paper presented at the annual meeting of the Association for Educational Communications and Technology, New Orleans, LA.

Super, D. (1984). *Mathematics activities courseware 3*. Markham, ON: Houghton Mifflin Canada.

Ubelacker, S. (1988). Elementary keyboarding is today's reality. *Canadian Journal of Business Education*. 1 (1), 24-29.

Paper

ITEC: An International Research Project on Children and Computers

Betty Collis
University of Twente, The Netherlands

Abstract

The ITEC Project (Information Technology in Education and Children) is a longitudinal, cross-cultural study of the relationships between different types of computer use, different types of school settings, different patterns of social interaction in the classroom, and children's higher-level cognitive functioning. The theoretical framework for this study is highly influenced by the theories of Vygotsky, which emphasize the importance of social interaction as an incubator for cognitive development. In this paper, we outline the background and history of the ITEC project, the theoretical model and various goals of the project, and some of the insights emerging from the 16 countries participating in the project. We conclude with a preliminary speculation on what the experiences of the project can offer to classroom teachers involved with computer use; to educational decision makers at the school, regional, and national level; and to researchers involved with the investigation of the impact of computer experiences in schools on children.

Introduction

Information technology, primarily involving the application of computers in the educational setting, is perceived by educators and the public in countries throughout the world as capable of having a significant effect on the developing child. The study of this effect— what mediates it and how it may be channeled to the best advantage of the child— is a topic of major interest to researchers, teachers, and educational decision makers throughout the world.

Despite this extensive interest in the use of computers in education, it remains difficult to draw conclusions from the experiences which are occurring. Partly this is because the field of information technology is continually changing in its characteristics; partly it is because "the effect" of computer use cannot be expressed in a straightforward fashion, but instead must be considered in the context of the complicated network of variables in which the use is embedded (Collis, 1988; McGee, 1987; Salomon, 1990). These variables include characteristics of the computer use itself, such as software types and design aspects, as well as hardware variations. They also include variables related to various characteristics of the students and teachers involved, and of teacher decision making, including how effectively the teacher integrates computer use into a meaningful learning experience for the child. The social interaction between student and student and between teacher and student also influences the impact of computer use in school (Feldman, 1989; Tharp, 1989). In addition, school, regional, and national culture also are part of the system of influences.

Despite these and many other problems involved in conducting credible research into the impact of computer use in schools on children, there is consensus that such research still must be done—but it must be done in a way that is sensitive to the system of variables that shape and are shaped by it. In response to this challenge, UNESCO, the Bulgarian Ministry of Science and Higher Education, and other sponsors representing researchers in 16 countries are collaborating in the "ITEC Project". ITEC stands for "Information Technology in Education and Children". ITEC formally began in May 1988 at an international meeting in Bulgaria initiated by UNESCO (although preparation for it occurred at another international meeting in 1987) and will finish its Phase 1 analysis and reporting by the summer of 1991. Sixteen countries have been actively involved throughout the study.

Conceptual Framework for the Study

The conceptual starting point for the study was derived from the theoretical work of the Soviet psychologist, Vygotsky. Vygotsky emphasized the important role that social interaction plays in cognitive development (Gallimore & Goldenburg, 1989; Vygotsky, 1978; Wertsch, 1985). Vygotsky hypothesized that an episode of cognitive development emerges through two cycles; "First, it appears on the social plane and then on the psychological plane. First it appears between people as an interpsychological category, and then within the child as an intrapsychological category" (Vygotsky, 1978, p. 163). A key resultant of this transactional, socially-mediated approach to cognitive development and learning is that such growth develops particularly well in the context of "every-day" problem solving interactions (Saxe, Gearhart, & Guberman, 1984). "Learning in context" is seen as the motive force for cognitive development, and thus the analysis of children's thinking and learning cannot be separated from the analysis of the social organization in which the learning transactions occur (Levine, 1988; Schoenfeld, 1989; Zimmerman, Smith, Bastone, & Friend, 1989). Vygotskyan theory, therefore, gave the project its rationale for the expectation that computer use in the social context of the school and cultural setting can affect the child's higher-level cognitive functioning (see also, Perret-Clermont & Schubauer-Leoni, 1989). Vygotskyan theory also underscores the necessity of studying the complex of variables pertaining to teacher and student interaction with regard to computer use in any consideration of the eventual effect of such use.

From Vygotskyan theory, therefore, and from the work of many other researchers studying the effects of computer use in a multidimensional social system (Clark, 1985; Collis, 1988; Eraut & Hoyles, 1988; Hawkins & Sheingold, 1986; Pea & Sheingold, 1986; Salomon & Gardner, 1986; and Watt & Watt, 1988; among others), the following clusters of variables emerged as a minimal subset necessary for an analysis of the computer-use/cognitive development relationship:

Background characteristics of the student, teacher, school, and culture, including:

- Family and neighborhood socioeconomic and educational levels
- Child's ability and achievement levels
- Child's prior experience, attitudes, and understandings with respect to computer use, to the content of the subject being studied through the computer use, and to the approach to learning involved in the computer use
- Child's gender
- Teacher's prior experience, attitudes, and understandings with respect to computer use and with respect to the organization of his or her classroom for computer use (see in particular, Hall & Hord, 1984)
- Teacher's preferred instructional style
- Availability and location of computers in the school, neighborhood, and culture
- School, neighborhood, and cultural "climate" relative to computer use
- Support and attitudes of the school principal relative to computer use and its potential in education

Characteristics of the computer use itself, such as type of computer application, design decisions in the software, and frequency of computer use

Characteristics of the social interaction surrounding computer use relative to student-student and teacher-student interaction (see Dalton, Hannafin, & Hooper, 1989; Grief, 1988), and

The instructional meaningfulness of the computer use.

Given this background, the ITEC Project takes as its model the multidimensional system shown in Figure 1:

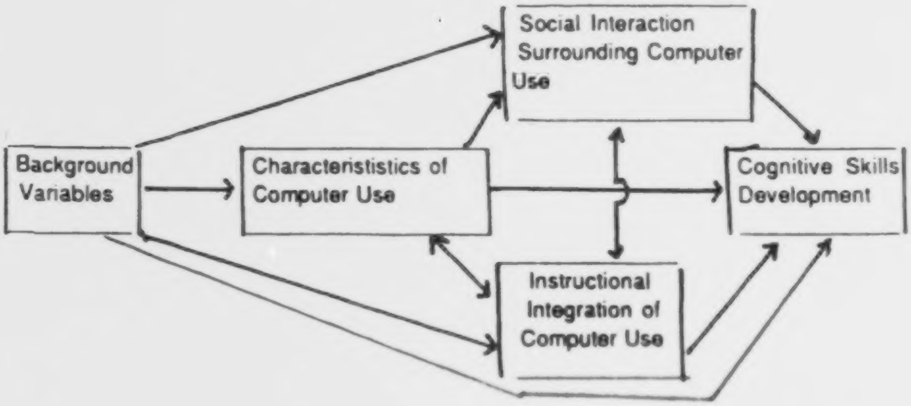

Social Interaction Surrounding Computer Use
Background Variables
Characterististics of Computer Use
Cognitive Skills Development
Instructional Integration of Computer Use

Figure 1. General theoretical model of the ITEC Project for the impact of school computer use on children's higher-level cognitive functioning.

and addresses the following research questions:

In the context of various combinations of background variables, under what combinations of

- characteristics of computer use,
- social interaction surrounding computer use, and
- instructional integration of computer use

is a positive impact on children's higher-level cognitive functioning more likely to occur? How do these sets of conditions vary in different cultures and countries?

In order to address these complex questions, particularly in a cross-cultural study, ITEC is divided into two phases, the first of which is focused on instrument development and cross-cultural validation, and also conceptual sharpening of what is meant by "higher-level cognitive functioning" in field settings. Thus Phase 1 of ITEC (1988-1990) is also addressing the following preliminary research questions:

- In the context of children using computers in the classroom, what are measurable or at least observable indicators of presumed "higher-level cognitive functioning"?
- Do these indicators vary cross-culturally?
- If the cross-cultural variation in the observable indicators is not too great, can a reliable methodology, usable in countries around the world, be found to measure their appearance and change over time?

Method, Phase 1

Subjects

Teachers. It was decided for Phase 1 to seek out existing classrooms in participating countries in which the teacher was known to be doing "good, interesting things" on a regular basis in terms of computer use in his or her instructional setting. We made this choice in order to maximize the chance of finding measurable effects on higher-level cognitive functioning. This decision was also made in order to somewhat control the variables relating to teacher experience with computers, teacher attitude about computer use and about innovation in instruction, availability of computers and usable software, school and community support for computer use, and experience of the students involved with computer use in an instructional setting. The participating researchers— 19 at the beginning of Phase 1 in 1988/89— made the selection of teachers based on their own local and national information sources. They obtained the consent of the teachers and the principals (and sometimes the parents of the children involved) concerning participation in the study. At the close of Phase 1 observations, Autumn 1990, 23 teachers were involved in the study: one each from Bulgaria, Canada, China, Hungary, Israel, The Netherlands, New Zealand, Sweden, U.S.A., U.S.S.R., and Zimbabwe; two each from Costa Rica, France, Japan, and Romania; and three from Mexico.

Although the teachers share the general characteristics noted above with respect to positive attitude with respect to computer use, they differ widely in terms of overall teaching experience (from 2 to 30 years, with a mean number of years of experience of 14); and in types of and length of training and experience with computer use. Twelve have taken formal courses, eight others have had various periods of training. Three have done teacher training themselves with respect to computer use. There is one more male than female in the sample of teachers.

Schools. Because of the Phase 1 procedure of looking for exemplary settings of computer use, it is not surprising that the schools involved in the initial sample have many exceptional characteristics, compared to other schools in their regions or countries. Seventeen of the schools have a special characteristic, such as having aggressively pursued involvement in a computer-related project, ability to convince others to equip the school with special levels of computer equipment, a special relationship with a university research department, or a principal with a particular vision for the school with respect to technology. Again, in line with our sampling strategy, we have tried to control these sort of contextual variables by including only (or mainly) schools that share these exceptional characteristics. In order to include countries with limited experience with computers, it is unavoidable that any school actually having a reasonable level of computer equipment and a teacher making regular use of it with his or her students is also an exceptional school.

Students. While the ITEC teachers and schools have exceptional characteristics, the participating children do not. It was decided to limit the study to nine- and ten-year children, as this cohort is old enough to make use of a wide variety of computer applications, but is still likely to remain with the same teacher most or all of the day and is still likely to be in a heterogeneous classroom setting, characteristics that are not possible after the age of about 11 in many countries. Approximately 680 children are in the ITEC classes; only one class, containing less than 20 of these children, was exceptional in terms of the class being comprised of deaf children. All other classes are described by their teachers as unexceptional for their neighborhoods.

While the children may be generally unexceptional, their amount of computer use experience generally is exceptional, largely because of the school and teacher sampling procedures used in Phase 1 of the study. What the children have done with computers varies through all the predictable possibilities— Logo and BASIC programming; applications software use; C.A.I of all types, including drill, games, simulations, music and drawing software; and use of Lego-Logo. Again, we wished as much as possible to observe children who were not too much distracted by early uncertainties with regard to computer use, but were more likely, through experience, to be able to use the computer in a variety of ways.

Procedure

After much discussion at a series of international meetings, it was decided that the observation procedure for Phase 1 would involve:

1. Preliminary interviews with the participating teachers and principals.
2. Description of the overall atmosphere of the school, its community, and the participating children. This was done through structured observations, interviews, and through a systematic videotaping procedure. These observations had to include a wide variety of examples of the typical types of social interaction among students and teachers in the school, away from computer use as well as around it.
3. In-depth interviews with the teachers and principals, relating to their opinions about the impact of computer use on their students, themselves, and their schools.
4. Profiles of each participating student, completed by his or her teacher, including information on the child's experience with computer use, ability level, work habits, learning style preferences, and many other salient variables.
5. A detailed description of the school's, teacher's, and students' experiences with computer use.
   (Seven detailed observation forms were developed to structure these observations, interviews, and videotape episodes.)
6. In-depth observation of at least three separate, full-length classroom lessons involving computer use of the type that the teacher felt would be likely to stimulate what he or she felt to be valuable higher-level thinking skills. For each observation, structured videotaping was done of the overall classroom environment in which the computer use occurred; other aspects of the lesson before, during, and after the computer use; and teacher-student and student-student interactions before, during, and after the computer use.
7. Following each in-depth observation, a detailed interview with the classroom teacher, in which he or she discussed the behaviors perceived to indicate higher-level thinking in the children, the strategies he or she used for computer use; his or her own appraisal of the impact of the computer-use session; and his or her identification of specific children, visible on the videotape or not, who seemed to be benefiting particularly well from the computer-use activities.
8. Following this interview, a written analysis by the researcher as to the extent to which he or she agreed with the teacher's appraisal of what occurred in the computer-use setting and with the teacher's appraisal of possible higher-level cognitive functioning among the students.

Analysis

During a number of different international meetings of the researchers, coding procedures were developed and validated for the videotapes, the interview data, and the various observation protocols. Over 60 pages of standardized coding forms are being used for the final synthesis of data. This synthesis was first done on a preliminary basis at the two international headquarters of the project, the Bulgarian Academy of Medicine and the University of Twente in The Netherlands (where the co-principal investigators of the project, Dr. Assen Jablensky and Dr. Betty Collis are based), and is being synthesized and interpreted by seven different groups of the ITEC researchers, each focusing on a different aspect of the analysis. These focuses are:

1. A final descriptive profile of the the participating schools, classes, teachers, and cultural settings
2. A final interpretive summary of principals' and teachers' opinions about the role and impact of computers in education
3. A summary of the computer-use situations involved in the three in-depth classroom observations
4. A summary of the teachers' and researchers' opinions and observations about the appearance of indicators of higher-order cognitive functioning (as triangulated by the videotapes)
5. An overall assessment of ITEC's findings and methodology relative to systematically cataloguing the system of contextual variables pertinent to an assessment of computer-use impact
6. An overall assessment of ITEC's findings and recommendations for further methodology with regard to the measurement of the dependent variable in the study—children's higher-level cognitive functioning.
7. An overall assessment of ITEC's experiences and recommendations relative to cross-cultural research in this area.

These analyses will be completed and will be in published form by May 1991. Presentation of the analyses by members of the ITEC research team had a major place at the Fourth International Conference of Children in the Information Age, held in Bulgaria, May 21-24, 1991. The presentation of the results at NECC '91 is the first North American discussion of ITEC Phase 1 insights and outcomes.

Contributions and Further Developments

ITEC has already made major contributions. Extensive pilot testing in many of the national sites; regular international working meetings in which scientists from the participating countries shared their ideas and experiences with respect to the impact of computer use in schools; an observation methodology which has proved to be manageable in an international setting and whose validity has been so far supported by our ongoing analyses of procedures; rich and fascinating videotapes (over 100 15-minute segments) of children using computers in real classroom contexts and visibly enjoying the use and their interaction with their peers during the computer use— all these are powerful contributions. Although it is premature to give "conclusions" of Phase 1 while analysis is still being done, it is fair to say the following:

1. Computer-use activities which the teacher and researcher feel have positive benefits for the children involved in the observations are not dependent on particular hardware or software characteristics or classroom environments. Enthusiastic engagement in learning, and presumed higher-level cognitive functioning, can take place where there is only one small classroom computer as well as in a lab of networked Macintoshes. Apparently beneficial classroom computer use not depend on doing a certain type of computer-use activity nor does it need some necessary baseline amount of computer access. Students interacting with each other, displaying enthusiasm in the way they interact with each other, and showing a willingness to sustain this quality of absorption with the learning task over considerable periods of time, also seem to be common characteristics, both of the context and of the higher-level cognitive functioning outcomes. The way the teacher introduces and organizes the computer use is important, the way he or she interacts with the children while the use occurs, and the way he or she attempts to reinforce and bring closure and transfer to the computer-use experiences after they are concluded also is important. This is the good news.
2. Measurement of higher-level cognitive functioning in the realistic social setting of the dynamic classroom remains very difficult to do. Teachers are convinced "good things are going on," the videotapes show the positive characteristics described above, but finding a cross-culturally manageable way to validly and reliably measure and document these "good things" does not seem to have emerged. Perhaps it is unfair to call this "bad news", but instead challenging news, in that our convictions about the powerful stimulating effect of classroom computers remain elusive to codify, quantify, and document. What we can say for sure is what we knew already— that "the effect of computers on children's higher-level cognitive functioning" cannot be considered outside of the social and cultural environments affecting the student— his or her home, community, the class, the teacher, the school, and the myriad interactions among these. And, perhaps as a corollary, that the enhanced social interaction among children which "successful" computer use frequently involves ironically makes cognitive measurement even more difficult at the same time as it makes the learning experience more valuable.

As we complete our analyses of Phase 1 we are also beginning our specific planning for Phase 2, which will involve a similar design as Phase 1, but with improved and expanded observations and analysis strategies. Announcement of plans for Phase 2 will occur at the Bulgarian conference in May 1991 and at NECC '91 in June 1991.

References

Clark, R. E. (1985). Confounding in educational computing research. *Journal of Educational Computing Research, 1*(2), 137-148.

Collis, B. A. (1988, July). *Manipulating critical variables: A framework for improving the impact of computers in the school environment.* Paper presented at the EURIT '88 Conference, Lausanne, Switzerland.

Dalton, D. W., Hannafin, M. J., & Hooper, S. (1989). Effects of individual and cooperative computer-assisted instruction on student performance and attitudes. *Educational Technology Research & Development, 37*(2), 15-24.

Eraut, M., & Hoyles, C. (1988). *Groupwork with computers.* Occasional Paper: InTER/3/88. The Research Consortium, University of Sussex and London Institute of Education, U.K.

Feldman, S. C. (1989, March). *An observational study of social processes in microcomputer classrooms.* Paper presented at the Annual Meeting of the American Educational Research Association, San Francisco.

Gallimore, R., & Goldenberg, C. N. (1989, March). *School effects on emergent literacy experiences in families of Spanish-speaking children.* Paper presented at the Annual Meeting of the American Educational Research Association, San Francisco.

Grief, I. (Ed.). (1988). *Computer-supported cooperative work: A book of readings.* San Mateo, CA: Morgan Kaufmann Publishers, Inc.

Hall, G. E., & Hord, S. M. (1984). *Change in schools: Facilitating the process.* Albany, NY: State University of New York Press.

Hawkins, J., & Sheingold, K. (1986). The beginning of the story: Computers and the organization of learning in the classroom. In J. A. Culbertson and L. L. Cunningham (Eds.), *Microcomputers and education. Eighty-fifth yearbook of the National Society for the Study of Education* (pp. 40-58). Chicago: University of Chicago Press.

Levine, H. G. (1988, April). *Computer-intensive school environments and the reorganization of knowledge and learning.* Paper presented at the Annual Meeting of the American Educational Research Association, New Orleans.

McGee, G. W. (1987). Social context variables affecting the implementation of microcomputers. *Journal of Educational Computing Research, 3*(2), 189-206.

Pea, R. D., & Sheingold, K. (Eds.). (1986). *Mirrors of minds: Patterns of experience in educational computers.* Norwood, New Jersey: Ablex Publishing Company.

Perret-Clermont, A. N., & Schubauer-Leoni, M. L. (Eds.). (1989). Social factors in learning and instruction. Special Issue, *International Journal of Educational Research, 13*(6), 575-580.

Salomon, G. (1980, April). *What is so special about computer-mediated communication? Studying the flute and the orchestra.* Paper presented at the Annual Meeting of the American Educational Research Association, Boston.

Salomon, G., & Gardner, H. (1986). The computer as educator: Lessons from television research. *Educational Researcher, 15*(1), 13-19.

Saxe, G. B., Gearhart, M., & Guberman, S. R. (1984). The social organization of early number development. In B. Rogoff & J. V. Wertsch (Eds.), *Children's learning in the zone of proximal development* (pp. 19-30). San Francisco: Jossey-Bass.

Schoenfield, A. H. (1989). Ideas in the air: Speculations on small-group learning, environmental and cultural influences on cognition and epistemology. In N. M. Webb (Ed.), Peer interaction, problem-solving, and cognition: Multidisciplinary perspectives (pp. 71-88). *International Journal of Educational Research, 13*(1), 1-119.

Tharp, R. (1989, March). *The institution and social context of teaching*. Paper presented at the Annual Meeting of the American Educational Research Association, San Francisco.

Vygotsky, L. S. (1978). *Mind in society: The development of higher psychological processes*. Cambridge, MA: Harvard University Press.

Watt, M., & Watt, D. H. (1988, April). *Re-examining the IBM Model Schools Program as a model for professional development*. Paper presented at the Annual Meeting of the American Educational Research Association, New Orleans.

Wertsch, J. V. (1985). *Vygotsky and the social formation of mind*. Cambridge, MA: Harvard University Press.

Zimmerman, B. J., Smith, C. P., Bastone, L., & Friend, R. (1989, March). *Social processes in microcomputer learning: A social cognitive view*. Paper presented at the Annual Meeting of the American Educational Research Association, San Francisco.

Projects
Networks

Implementation and Evaluation of an Electronic Network for Teachers and Administrators

Ann Thompson
Julie Hamilton
Iowa State University

Abstract

Abstract unavailable at press time.

Using Computer Conferencing to Bridge the Gap Between K-12 and College

William W. Wright, Jr., Director, BreadNet/ITC (International Telecomputing Consortium)
Washington, DC

Abstract

In the educational telecomputing world, there is much talk these days about legislation in Congress calling for the support of a National Research and Education Network (NREN). Mike Nelson, staff person for U. S. Senator Albert Gore, Jr. (who sponsored the legislation) says that while the NREN bill will support researchers who need a multi-gigabit network, there will be other bills so that the K-12 world can connect.

EDUCOM, a not-for-profit, Washington, DC-based consortium, which has traditionally worked with those using technology in *higher* education, has created a position for a K-12 specialist. This person will start by collecting information about K-12 projects that are working on BITNET, an electronic mail system that university-level people use at no cost.

The National Science Foundation (NSF) is looking at how a national network for K-12 might take shape. Many of the regional or mid-level networks of NSFNET are beginning to support the K-12 world.

Some of these projects are charging ahead toward universal connectivity without looking at ways that teachers have organized themselves on networks since the early 80s. I will talk about a network of teachers called BreadNet which was initiated in 1983. K-12 teachers who attend the Bread Loaf School of English, one of the ten summer graduate programs of Middlebury College in Vermont, use the network during the school year to share ideas and set up projects for students. After the second year we moved from e-mail to computer conferencing, a much more dynamic environment for group work. I will tell about recent projects that:

- let K-12 classes of Native American students work with university classes at Georgetown University and Berkeley
- let high school students have a year-long (online) discussion with college students about how to prepare for college
- let college writing instructors share ideas with high school teachers about what works (I will also tell about problems of a "gateway" between higher education teachers on BITNET and rural K-12 teachers who did not have this resource.)

I will talk about what we learned from these three efforts and from six years of shaping an online community. For example, we think that working in small online groups on a computer conference provides more benefits than a one-to-one "pen pal" exchange via electronic mail.

The IBM/NEA School Renewal Network: Linking Practitioners and Researchers

Shari Castle
Coordinator of Research and Develpment
NEA

Abstract

The IBM/NEA School Renewal Network uses a horizontal dissemination model that helps solve many of the persistent research-practice interaction problems. Practitioners from 50+ sites who are actively engaged in school restructuring and 11 researchers from universities, regional labs, and research centers interact on ten topics: Thinking, Instructional Strategies, School /Classroom Organization, Curriculum Involvement, Networking / Technology, Restructuring, and Student Assessment. This interaction enables access to research, the use of the formulation of contextually -sensitive solutions to complex schooling problems. Telecommunications makes this horizontal interaction possible.

The presentation will focus on the purpose and "spirit" of the Network by using actual samples of Network conversation.

The School Renewal Network is partially funded through a grant from the Secretary's Fund for Innovation in Education, US Department of Education.

Projects
Gender Equity in Computing

Using Hypermedia to Bring Young Women to Science

Rachelle S. Heller, C. Dianne Martin, and Emad Mahmoud
Electrical Engineering and Computer Science Department
The George Washington University, Washington, D.C.

Abstract

The "Bringing Young Minority Women to the Threshold of Science" program, sponsored by The George Washington University and the National Science Foundation, takes a unique approach in drawing young minority women to computer science, engineering, and science careers. Each year of this project 24 minority female students from the 9th and 10th grades selected from the public and independent schools in the greater Washington, D.C. area are given the opportunity to use the latest computer technology to engage in scientific inquiry in cooperative teams under the guidance of computer science professionals and science teachers. So engaged, these young women are learning about both science and computers, but more importantly, they come to view themselves as potential university students and scientists.

The Portrayal of Females in Computer Education Texts Grades K-12

Gregg Brownell
Bowling Green State University, Ohio

Abstract

Studies suggest that females' attitudes toward computers are less favorable than males' attitudes. This may be due to the cultural stereotyping of computers as a male domain. One area where educators can have some control over this stereotyping is in the representation of women in the materials that students view. This study investigated 23 computer education texts ranging across grades K-12. Two raters examined each text and noted each illustration that portrayed a person or identifiable part of a person. Each illustration was coded as a single male, a single female, a group of males, a group of females, or a group of males and females. If the illustration contained a group it was coded as to who was in power. Each individual was coded as adult or child and as to role. Lastly, each picture with a computer was coded as to active user, passive participant, or rejector of technology. Chi square tests were performed to evaluate differences between 1) the observed and expected numbers of men, women, boys and girls, 2) men and women regarding role portrayal, and 3) men and women in their relationship to the computer. The results of the study will be presented.

Calculating Women: Filling in the Time Line

Marcelle Bessman
Associate Professor, Mathematics
Frostburg State University

Abstract

In 1989 IBM sponsored a competition for its multimedia product, The Audio Visual Connection (AVC). I received one of 36 awards of the AVC for my proposal "Women, Science and Society: Women of Mathematics". An objective of this proposal was to develop an interactive, multimedia instructional program that would facilitate study of participation of women in the development of mathematics. Each woman would be viewed within her historical, social, political, psychological and esthetic context as well as with respect to her professional work and proper status.

In the occurs of working on my project I renamed it "Calculating Women: Filling in the Time Line". The "Time Line" refers to a mural entitled "*Men* of Modern Mathematics" that I use as a vehicle for moving through the history of mathematics. Through multimedia techniques *women* mathematicians are being added.

This year marks the 100th anniversary of the death of Sofya Kovalevsky, the first woman to achieve a Ph. D. in mathematics. This multimedia presentations is intended to honor her by focusing on her place in the "Time Line".

The status of the program and my plans for its future will be discussed.

Projects
USA Students Exchange with Israel, Haiti, and USSR

Tall Tales: Intercultural Creative Writing Exchange Between Russian and American Students

Anna Gritsouk
Michael Waugh
University of Illinois

Abstract

This project involves a creative writing exchange between students in the United States and the Soviet Union using the PeaceNet computer network. It is a part of the New York State/Moscow School Telecommunications Project jointly sponsored by the Copen Family Foundation and the Soviet Academy of Sciences.

The teachers involved were Mr. Robert Korthas, an English teacher at the Sacandaga Elementary School in New York, and Ms. Anna Gritsouk, an English teacher in School #34 in Moscow. The goals of the project were jointly determined in a planning conference in Moscow during the summer of 1989. The project's primary goals were to help increase the interest of Russian students in learning English, to help them develop their imagination and creativity, and to help them develop greater ability to think in English. The project's secondary goals were to help the students gain information about each others cultures and everyday lives.

Learning activities on the Russian side included translating and publishing the Tall Tales stories in their own language, writing Tall Tales of their own in the manner of American Tall Tales, and performing a play based on the American Tall Tales.

The project proceeded in several distinct phases, from learning about telecommunications and developing procedures for generating and receiving electronic mail, through the actual exchange of information about Tall Tales and the exchange of collaboratively written and individually written Tall Tales.

This presentation will present a summary of how the activity was conducted and assessments of its impact on the Russian students English language skills. It will also describe lessons learned on the Russian side concerning the difficulties encountered with conducting activities of this type using electronic mail and plans for future activities.

Putting Life into Literature Via the International Teleclass

Judith Eber
GesherNet
Heschel Day School, Los Angeles, CA

Abstract

An intercultural encounter between students in an Appalachian community in Kentucky and students from a kibbutz settlement in Israel evolved from a simple exchange of information to discussions of jointly read works of literature. The computer conferencing format on GesherNet (an Israeli/American educational network) enabled students and educators to exchange thoughts, opinions and ideas in an extended classroom group beyond each of their physical school environments. Students gained new perspectives on their own Interpretations of literature as well as a deeper appreciation for one another's societies, histories, and cultures.

The Literature Teleclass on GesherNet for the 1990-91 academic year grew out of this enriching literature exchange. Ironically, only briefly into a joint discussion of Shirley Jackson's "The Lottery," between Israeli and California students, Saddam Hussein seemed to have selected himself to determine the fate of the Israelis by launching Scud missiles on the country.

Telecommunications became the "lifeline" enabling students to relate first hand experiences, express concern, share prayers, show support and debate issues related to the Persian Gulf Crisis. "Living" history pre-empted literature.

Through such Teleclass experiences—a glimpse at student messages, the development or abandonment of lesson plans, and the recognition of distinctive characteristics of telecommunications—the components for building a comfortable "telecommunity" for literature (and "life") exchanges are revealed.

So What Can You Do With Technology and Second Languages?

Ivy J. Montoya
University of Miami, Florida

Abstract

This project presentation is the result of two years of computer-infused instruction in second language learning. Although the curriculum was designed using the basic principles of sound instructional design, some activities were significantly more successful than others. Teacher observed results of achievement, motivation and affect will be outlined along with each activity. Included in this presentation will be the use of computers to promote computer literacy for at-risk foreign language students, activities for the word processor, database and spreadsheet, and computer-generated, student-produced video productions. A method of using *HyperCard* stacks to reinforce second language instruction will be demonstrated, using the scanner, and MacRecorder for aural reinforcement of the new language. A telecommunications project plan using AppleLink to translate health occupations terminology to Spanish for schools with no hispanic population will be summarized. A *HyperCard* stack will preview Miami Central Senior High's 1990 Apple Crossroad Grant. This integrated curriculum project is directed at ESOL students to assist them in making the transition into our South Florida Community before they have learned English. It is to be tri-lingual in format, and will have capability of expansion to include other languages.

Special Presentation
Long Distance Learning and Computers

Tom Campbell
Cochise County School District
Brisbee, Arizona

Abstract unavailable at press time.

Special Presentation
Federal Perspectives on Technology and Restructuring

This panel will discuss federal perspectives on technology and restructuring. The set of presentations will encompass alternative views of what restructuring means (both for education and for society), technology's role in its evolution, and its potential national implications.

Advanced information technologies are changing the knowledge, skills, and values needed to be a worker, citizen, and self-actualized human beig. In a global marketplace, for America to regain prosperity, developing an accomplished and motivated populace will be at least as important as technical advances, partnerships between government and business, international treaties, and decisive leadership. The quality of the nation's educational system is central to preparing us to best the challenges we face face on the brink of the millennium.

Many have documented that present approaches to schooling are deeply flawed and should be radically changed. In light of current understandings about learning, organizational development, and knowledge representation, almost every assumption underlying the current educational model is obsolete. Shifting to a new paradigm for teaching/learning is crucial to developing human resources for transcending our present malaise.

The use of advanced information technology is essential to the success of school restructuring. Individualized learning and decentralized institutional structures require complex organizational practices that necessitate sophisticated computational and communications capabilities. Stepping into a classroom should be like entering a time machine hurtling forward; today's educational system should foreshadow the intelligent tools and interactive media that will pervade future workplaces and communities.

Panelists:

Chris Dede
George Mason University
Fairfax, Virginia

Beverly Hunter
National Science Foundation
Washington, DC

Linda Roberts
Office of Technology Assessment, U.S. Congress
Washington, DC

Special Presentation

Interactive Multimedia in the 90s: Developments, Trends, Prospects, and Applications

Education in the 90s could be vastly affected by interactive multimedia platforms that extend what computers can do by adding color, sound, video, animation, and images to interactive computer based simulations and dialogues. Members of this panel have all been involved in the development of multimedia products and they will focus on some of the issues in the design of interactive multimedia. Using presentations, they will discuss design standards and aesthetics involved in merging multimedia with creative instructional design. Demonstrated products will form the basis for a discussion of instructional issues. Panelists will also attempt to forecast the future of interactive multimedia in an educational setting and establish some recommendations for future development.

Panelists:

Terry Cannings and Linda Polin
Pepperdine University & Interactive Multimedia Inc.

Peter Kelman
Scholastic Software

Steve Arnold
Lucasfilm Learning

Greg Laskaris
Discovery Channel

Perry Reeves
Pioneer Communications of America

Special Presentation—ISTE
Out of the Trojan Horse — Logo's Role in Educational Reform

In his 1986 paper, "Logo as a Trojan Horse: Rethinking Logo Philosophy in the context of a Real School Experience, " Seymour Papert stated the following;

"...The central question for educators is whether schools of the future will go on teaching the same curriculum, using computers to do the job better, or whether we'll see radical change in what is taught and learned in schools. Ishal suggest that the education system will not be able to bring itself to decide on radical change in education. But their will be a radical change all the same, despite the system's lack of decision making power. change will come about through what I call "The Trojan Horse Effect"—the system will admit what look like innocuous materials which will serve as vehicles for "germs" of change."

This panel discussion will attempt to determine Logo's contribution to educational change. we will address many complex questions. Has Logo been successful in it's goals of empowering children, unifying knowledge, creating a culture of life-long learners, and restructuring schools? which educational trends have influenced the continuing wide-spread use of Logo in schools? What kinds of alliances should the Logo community be forming with the broader educational community and how? What does Logo have to offer the traditional educational establishment? What substantive changes need to be made before the Logo philosophy can become widespread?

Was Seymour Papert's 1986 prediction accurate? should we as a community of educators be more honest in our goals and inspirations? Is it time to come out of the Trojan Horse and declare what we stand for educationally?

I am optimistic that this session will illuminate some important global educational issues and will influence the ways in which many of us think about educational technology.

Participants:

Gary Stager, Chair, SIGLogo
Douglas Clements
Marion Rosen
Michael Tempel

Projects
Computers and Girls

Where is Carmen? In the Classroom.

Mary Jane Ford
University of Southwestern Louisiana

Nancy Markwell
Episcopal School of Acadiana

Abstract

The presentation will describe a project designed to encourage computer use by seventh and eighth grade girls. Girls in the seventh and eighth grades at a private coeducational middle and secondary school in a small Louisiana community are required to take computer literacy courses but express little desire to continue with computer activities. In an effort to include computers in their instruction and to increase their use of and interest in computers, an enrichment course designed to use the Carmen Sandiego series of software will be offered in the spring quarter of 1991. The course is being designed with the cooperation of several subject-matter teachers so that the use of the software will be integrated into the curriculum. Students enrolled in the enrichment course will be given pre- and post-attitude surveys to determine any effect the course has on attitudes about computers and their use. Other data will be collected to determine any changes in achievement in the various subject areas involved.

This presentation will describe the enrichment course provided for students, the activities used in the course, and the analysis of the data collected.

Equity in Technology: Removing the Barriers to Equitable Computer Access

Alice Fredman
ECCO

Abstract

With the introduction of computers into the classroom, groups that tended to suffer past educational inequities now suffered one more—the technological inequity. The students most affected by these subtle forms of discrimination fall into four categories: females, economically disadvantaged, minority, and handicapped. While the benefits of computers continue to multiply for many of our students, these four groups are in danger of becoming the victims of a new kind of segregation which separates the computer literate from the computer illiterate.

ECCO's Equity in Technology Project was designed to help teachers identify areas of inequity and to devise ways in which to address and alleviate them. We believed that the classroom teacher was in the best position to observe how technological inequities were manifested and how they might be overcome. Self-image, perceptions by others, improvement in academic areas, and increased comfort with sophisticated technology were important outcomes of the projects, particularly so since our target populations were often denied opportunities to achieve these kinds of successes.

This presentation will provide an overview of the project, discuss themes common to each population, and suggest ways in which inequities can be overcome.

Girls and Computer Technology—Maintaining an Interest

Joyce Kasten
Trinity Center, California

Abstract

Abstract unavailable at press time.

Special Presentation—ISTE

Technology Makes a Difference: Improving Our Schools

Lud Braun
Dix Hills, New York

The International Society for Technology in Education(ISTE) recently concluded a study of the effectiveness of technology in improving the education of young people and in assisting restructuring efforts. The major findings of the study are:

- Technology can help well-trained and well-equipped teachers to develop the intellectual potential of all their students.
- Technology in the hands of trained teachers is especially powerful when used with at-risk students
- Teachers are the key to success in any educational change. They must be trained in the change. They must be supported in implementing that change. They must be provided with adequate resources to create the change.
- Permitting continuing existence of dropouts and inadequately-prepared graduates is very costly both in loss of human potential and in productivity of a country's citizens. If account is taken of the financial loss from these poorly prepared students, money can be found to fund the introduction of technology into classrooms in numbers which will make a dramatic difference.
- Business people and educators have much to offer each other in approaching solutions to the problems faced by schools and by students.

Among the matters presented will be evidence of effectiveness of technology in the classroom (including description of case studies), structural changes required in the educational system to take advantage of technology, and the financial model of economic impact of dropouts.

Projects
International Computing: Mexico

Teaching High School Physics Through the Use of Computer Simulation Laboratories

Enrique Calderon

Abstract

Physics is regarded as an important but difficult subject by most high school students, yet it's importance in today's world is perhaps most relevant than ever, a thorough way to understand the full significance of technology and to enjoy and care the magnificent universe we live in. Basic to physic understanding and insight, is experimentation, as opposed to books or movies, equivalent to acquire secondhand knowledge. Experimentation is also an alternative to the use of mathematics, trough which the learning of physics becomes cumbersome and obscure. Until now, experimentation has been very limited to high school and even to college students, because of the technological and financial problems it posses.

The use of computer simulation models of physical systems would greatly improve the access of student to real life type laboratories, making it possible to carryout experiments with complex phenomena, like gravitational field dynamics, mechanical waves, energy transformation, or mechanical stress in bridges and towers.

This paper describes an experience performed in the Mexican Galileo Project, in which students are currently working with several physics laboratories, while developing new ones.

The paper includes a few aspects of the computer models relevant to learning.

On New Tools to Close the Gap Between Between Mathematics

Gustavo R. Deffis

Abstract

While some adults and quite a few students love mathematics and enjoy working with it, the great majority of them have a negative feeling about it, which goes from indifference to repulsion all the way. The root of such feelings lays in the way they were taught math, and faced mathematical objects, like equations and functions. Today a new perspective is opening in which the treatment and study of those subjects and the relation among them becomes easy as it should be. Mathematical objects are very simple ones, perhaps the most simplest ever, but they become complex and inaccessible when we try to compute them by hand, loosing time and the possibility to see the whole picture, while we repeat mechanic operations over and over again.

Until now, teachers have had no other way to teach mathematics, but the hard one. If someone is to learn the parabola, he requires to work with the equation to compute and draw the curve to visualize it. A few exercises will teach him both, the nature of this objects, as well as, to respect them as a family of ugly beasts.

The work reported on this paper describes a bunch of tools integrated in a chest, conceived to simplify the life, and to ensure in the student, a feeling of friendliness to analytical geometry and differential calculus objects. The basic idea is to let them at its own with all sorts of functions and equations looking at the forms they produce, at the same time, the tools should be powerful enough to solve any of the problems students usually find in the courses they enroll. A discussion of some results founded so far is also included.

Logic and Creativity Experiences at Kindergarten Level Using the "Magic Microwave Oven"

Claudine Court

Abstract

This work reports and evaluation experience of the software program "The Magic Microwave Oven", performed with students from kindergarten in Mexico city schools. With this program, tiles are created when geometrical shapes of different colors are placed one over the other. With this pieces, pictures can be assembled, in away similar a jigsaw puzzle.

This program has been more widely used with students at elementary level and it has been observed that abstract planning of actions, logical thinking and visual perception analysis, takes an active part in the scheme.

At kindergarten level, an individual trace was held in order to observe which activities children at his age carry out, also performed by children in elementary school up to a certain extent, but when solved easily, they are difficult to observe. Several

response levels were found and they correspond to different situations that the student has to solve when he uses the "Magic Microwave-Oven"

Some of the situations are: -Identification of colors. -Perception of one color as being the same one, independently of the illusions created by contrast of colors. -Classification of geometrical shapes. -The importance of order in the placement of shapes and its effects when it is modified. -For design or drawings: Abstraction of the shapes and adjustment to geometrical shapes included in the software package. -Management of spatial relations on the plane. -Parts-whole relations, as those found on jig-saw puzzles.

Discovering Language Structures

Rocio Gonzalez

Abstract

This work reports and individual evaluation experience using the software "Finding it". The students are currently at elementary level in Mexico City. This software has a similar format to that of "closed tests", where a whole word is removed as many times as it is found within a text, that the user will have to complete during the test. Using the software, words can be removed, as well as any other element such as accents, punctuation marks, vowels, etc. When the student's performance was analyzed, the following was found:

- Generally it is easier for the students to complete a text where vowels are missing, than those with missing consonants or missing whole words.
- Even the youngest students give answers that agree with the grammatical category of the text.
- To solve each exercise, the students read again the part that has been solved each time they are not sure what follows. It is evident that they need to understand the text in order to continue.
- The youngest ones, don't "expect" specific grammatical categories within the text. For instance, if it is written "A ——" they don't try different nouns, rather, they try letter by letter. On the other hand, students around ten years old, in this cases search according to grammatical categories, and think of answers such as "A cat", "A mouse", "A pencil" etc.
- When the structure suggests the search for the right word, and not letter by letter, as it is pointed above, there is an evident change of strategy by the student.
- Those students that discover some rules of the language fill very excited and motivated to search for more. They have a pleasant surprisewhen they findout that language is regular in some aspects. This refers to eleven to twelve years old students.
- When students are sure of the word, but they misspelled it, and as a consequence the program does not allows the to write, it is frequent that they try other ways to write it.
- Students like this software and find it interesting excluding those cases when the only major contact with a computer is by using action games.

Special Presentation—ASCUE

How to Approach Knowledge-Based Systems: PROLOG or Expert System Shells?

Waldo Roth
Computing and System Sciences Chair
Taylor University, Indiana

Antonio Lopez
Loyola University of New Orleans
Louisiana

Current education in Knowledge-Based Systems will be reviewed as it relates to both pedagogical and industry needs for the two approaches of logic programming vs. expert system shells. Both speakers have extensive experience and educational successes with one or both of these technologies.

Dr. Lopez has recent summer experience working with NASA as a college faculty member which may also be of interest to listeners.

Special Presentation—SCS

Mathematics Exploration and Experimentation: A Technology Based Teacher Development Course for Inquiry Teaching and Learning

In this session we will describe our experiences using a course developed under an NSF grant to help middle school teachers learn to use technology to support inquiry learning in mathematics. The course used technology based activities to involve teachers in their own mathematical inquiry learning and then used those experiences to help teachers reflect on how they might do inquiry based activities in their own classrooms. Topics for inquiry included geometry, number theory, fractions, functions and graphing in pre-algebra. We will share some of the development that resulted in teachers classrooms and discuss steps and pitfalls in using technology to help teachers make the transition from traditional to inquiry oriented mathematics as advocated in the NCTM standards.

Projects
Hypertext/Hypermedia in Content Areas: Music and Business

The FoxWay Project: Hypertext v. Database

Deborah Healey
Oregon State University

Mark Horney
University of Oregon

Abstract

The FoxWay project is developing the uses of hypertext and database applications in education. The project demonstrates that different computer applications—specifically, hypertext and databases—can allow students multiple interpretations of the information they collect.

We took the *Spoon River Anthology* and converted it into a hypertext document and into a data file for a database. We then conducted a parallel analysis of the poems in the anthology using the two different computer programs. With the database, we performed a content analysis. With the hypertext, we made structured overviews of the relationships between characters and concepts contained in the poems.

During this process, we made the following observations: Due to easy finding, sorting and filtering, the database encouraged counting operations. The hypertext facilitated browsing and "fuzzy" connections among ideas where one poem could be linked to another without specifically defining the relationship between them. The inherent metaphorical structures of each program predisposed the students to view, manipulate and interpret the data in specific and different ways.

Computers change the nature of how data are interpreted, and different programs provide different points of view. Thus, the use of both programs in the classroom provides opportunities for multiple insights.

Integrating MIDI, Videodisc and HyperCard as a Music Educational System

Clinton E. Stapleton and James A. Levin
University of Illinois at Urbana-Champaign

Abstract

Though there are a large number of computer systems for music instruction, few offer the ability for relatively simple integration of hardware and software that can be edited by teachers or students. *HyperCard* is a low cost, easy-to-use Macintosh application which offers a way of controlling external hardware such as MIDI, CD-ROM, and videodisc players. *HyperCard* allows users to create instructional software or modify existing software.

The purpose of this study is to investigate the feasibility and effect of integrating a MIDI keyboard instrument and videodisc player into a *HyperCard* stack as a Dynamic Instructional Music Environment (DIME) for delivering music instruction to precollege and college students. This study examined the following questions: Will students with a beginning to intermediate level of music experience be (a) motivated to learn music theory knowledge and skills, (b) able to demonstrate music theory knowledge and skills by constructing and performing perfect octaves, major thirds, and perfect fifth intervals on a piano keyboard, (c) able to demonstrate transfer of music theory knowledge and skills by constructing and performing major thirds and minor thirds on a piano keyboard, (d) able to demonstrate music knowledge by discriminating musical instruments according to wind, percussion, and string as a result of using DIME?

A mixed population of subjects were selected for the study and an attitude (Likert) scale and one group pretest/posttest research design were employed. Results from the pretest, posttest ,and Likert scale suggested that DIME contributed to: (a) successful performance of perfect octaves, major thirds, and perfect fifths, (b) transfer of learning, since students were able to solve

related problems of constructing and performing major and minor triads using prior knowledge and skills about music intervals, and (c) positive attitude towards learning music with DIME.

This study demonstrated that it is possible to extend the audio and visual capabilities of the microcomputer for music instruction in an inexpensive manner and that high fidelity video and audio may be integrated by music teachers and students to design their own computer-assisted instruction, with an easy to use and edit software authoring tool.

Hypertext and Professional Communications: Developing Design Principles

Emily A. Thrush
Memphis State University

Abstract

Professional and technical writing courses teach future engineers, scientists, and writers principles of document design based on research on how readers read. The growing use of documents meant to be read on a computer screen has prompted studies on those elements of design and usability peculiar to this environment.

Hypertext, a relatively new application of computer technology, now poses new problems in document design. In a sense, two-dimensional documents become three-dimensional in hypertext, because the reader can delve into the material at varying depths. This places new demands on both the designer and the user. While a great deal of study and planning has gone into the architecture of hypertext systems, little study has been done on how the document should *look* on the screen to be easily accessible, efficient, and effective.

The purpose of the HyperScreen Design project at Memphis State University is to develop principles for hypertext documents. First, design elements were isolated through an analysis of hypertext documents and the options offered by hypertext authoring systems. A sample document has been prepared with variations in design elements for a research study to measure the effects of the variations on usability.

Special Presentation
Media Issues in Telecommunication

Marina Stock McIsaac
Arizona State University

This symposium will explore the following issues in the area of telecommunications:

1. Distance education for professional and continuing education (Vicki Bagget)
2. Retrofitting of classrooms for broadband and online video (Rob Fidler)
3. Computer conferenceing (Phil Konomos)
4. Video Technologies (John Barnard)
5. Still visual technologies (David Ragsdale)
6. Distance learning and virtual reality in the 21st century (Steve Hustedde)

Paper

Evaluating Students' Use of Electronic Communication to Build Consensus: A Pilot Study

Ann L. Lyness
University of Pittsburgh School of Nursing

Jean A. Raimond
University of Pittsburgh School of Nursing

Abstract

An evaluation pilot survey was administered following use of electronic communication by doctoral nursing students to practice consensus building skills. Building consensus electronically was designed as a teaching strategy to promote practice in synthesizing and exchanging ideas within a technological learning environment. Incorporated into a seminar course that focuses on structuring nursing knowledge, the strategy involved readings that address different issues in making knowledge more useful to people. Students could use electronic mail to communicate to the other students in the class on the content of the readings. Instructor support made possible the posting of the student messages on a local electronic bulletin board developed for the course. Results of this strategy were positive in that students began building consensus. Ratings were high overall on the evaluation survey. Students expressed satisfaction with the experience but also suggested that more time be allowed to become competent with the EMail and bulletin board systems.

Background

Course Emphasis. The course "Seminar on Structure of Nursing Knowledge" has been offered at the University of Pittsburgh School of Nursing to doctoral students since 1988. Designed to be taken after most of the course work was completed, this seminar required students to develop a knowledge structuring project that organized theoretical, research and other information meaningfully. Part of structuring included learning how to provide relevant synthesis products such as summarized reviews for nurses in practice. Activities were introduced that gave the student an opportunity to synthesize ideas. Since striving for a consensus often involves synthesis reasoning, a consensus building assignment was developed for the course.

Consensus Building. As both a scholarly and practical process, consensus building is usually accomplished in phases. Initially, one or a few experts synthesize published knowledge within a focused area. Next, agreement is sought with other experts concerning the combined information. The first phase emphasizes development and consists of the following: a) review of literature, b) identification of key concepts and relationships, c) analysis of the literature to bridge gaps or conflicts, and d) identification of implications for practice and policy making. The second phase emphasizes consensus seeking and reaching as the developed synthesis product is revised by more experts. Synthesis occurs frequently during the developmental phase. Less extensive synthesis occurs during the seeking/reaching phase (Ward, 1983a; Ward, 1983b). One of the practical outcomes of consensus building is that it can help unify complex research work by bringing about understanding and a greater degree of agreement in areas of difference.

New Teaching Strategy. Within a seminar discussions surrounding problems are expected. A seminar provides for looking at problems and includes guided discussions toward solving the problems. For this seminar, making knowledge useful to practitioners was identified as an important problem to address. Consensus building showed promise to sharpen students' thinking about problems related to knowledge use. Several required readings dealing with knowledge utilization presented their views differently. These readings served as the basis for a small consensus building assignment aimed at identifying implications for knowledge use in practice. Students could exchange their ideas while seeking consensus. It was felt the consensus building experience would strengthen the students' ability to synthesize knowledge, increase leadership skills through initiated communications with peers, and gain awareness of factors related to the use of knowledge. Since this was a course at the doctoral level, an effort was made to incorporate a future oriented aspect to the teaching strategy. Electronic communication using EMail and a local bulletin board for consensus building, was selected to offer a new type of communication experience so that students not only could exchange ideas, but they could also gain insights about the electronic aspects for future use.

Electronic Communication

Electronic communication means the transmission of information as a result of electronic equipment and software, most often computer technology. Common forms of electronic communication include electronic mail (EMail), bulletin board systems (BBS) and search databases. Electronic communication offers a new convenience for communicating. For example, electronic mail permits exchanging messages via phone modem or through direct cabling (Staggers, 1989) via terminal emulation software. The modem and emulation software are used to convert the source computer's digital data into audio tones (analog data) that can be carried over telephone lines. This conversion is called modulation. Once the data reach the target computer, the modem converts the analog data back to digital data. This conversion is called demodulation. The word "modem" reflects both functions of modulation and demodulation. Modems are classified according to their speed or bps (bits per second). Currently, 2400 bps is standard, with 9600 bps now available. In the future, electronic communication will travel over the Integrated Services Digital Network (ISDN). The ISDN is essentially digital telephone links, capable of 64,000 bps communication (Dvorak & Anis, 1990). The ISDN will eliminate the need to convert data for transfer over phone lines and will provide end to end digital connectivity (Wigand, 1988).

Through electronic mail, messages are relayed to the recipient's electronic mailbox shortly after they are sent. This means that messages can be sent and received within minutes and at the sender's and receiver's convenience. With the amount of the world's information growing so rapidly, access to information for timely decision making is crucial. EMail provides the user with the power to access the information. EMail systems may be categorized by access potential and by distance. Some systems, such as private EMail systems operate on Local Area Networks (LANs). These systems usually have very limited access, perhaps to members of only one institution. Other EMail systems include public systems such as AT&T, MCI and Western Union's Easy Link. Semi-public systems such as CompuServe offer access to the public at large by subscription. Bitnet is a university related system that offers international capability. Directly wired computers tend to communicate locally within a short distance as within a hospital or a university campus.

As the volume of information being channeled electronically has increased a natural next step was to create intersections of information such as bulletin boards. An electronic bulletin board system is a centralized data storage network designed for organized access by multiple users. Some bulletin boards are interactive permitting input and editing of communication by users. Some bulletin boards contain "read only" databases. At present there are more than 3000 databases of information online and over 10,000 public electronic bulletin boards (Dvorak & Anis, 1990). Services provided by BBSs include information from newspapers, magazines, journals, organizations and job recruitment services. In nursing, examples of BBSs and databases operated by nurses include: Nightingale BBS, Nurse Link, Nurse Corner, SON*NET, Health Link (Skaggs, 1990) and the International Nursing Library of Sigma Theta Tau (Hudgings, 1990). These BBSs and databases can be accessed across the country. The BBS has been researched for interest to continuing education providers in nursing. In Florida, a survey concerning use of the bulletin board for continuing education, was sent to a sample of 225 continuing education providers approved by the Florida Board of Nursing. The findings suggested that while support for this format in Florida may be emerging, start up and maintenance costs, and lack of preparedness of the nursing profession were obstacles (Russin & Davis, 1990). For nurses to benefit from electronic communication, they have to have the appropriate equipment, software and knowledge to conduct work within an electronic communication environment. By providing students an electronic communication experience, they are better prepared for future work endeavors.

In a Speech Communication Project at the Pennsylvania State University, a traditional group discussion course was redesigned to operate via computer mediated communication. Findings indicated more frequent student contact with instructors, high level of course approval by students, and group output that equalled or excelled that of the traditionally taught course (Phillips & Santoro, 1989). For consensus building, it seemed appropriate to use electronic communication to increase dialogue and facilitate sharing of information and knowledge.

Pilot Study

Introduction. When little is known about a new area of study, pilot research is a useful way to begin (Polit & Hungler, 1983; Roberts & Burke, 1989; Wilson, 1989). The pilot study is usually done on a small scale, provides baseline information and may help to uncover snags that could interfere with carrying out a larger project. It may show aspects related to cost-effectiveness, time, and cooperative arrangements needed among different groups of people. Revisions in procedures or the instrument can be made after the pilot study has been conducted.

Instrument. The instrument, an evaluation survey, was developed as a Likert-type instrument primarily but also had several open ended questions (see appendix). The Likert scale was originally developed by a social psychologist Rensis Likert to study opinions or attitudes of people (Burns & Grove, 1987). The instrument consisted of a series of statements to which the subjects responded on a scale. Over the years adaptations to the original have been made by various researchers. Odd or even numbers of choices have been used. The even number has been referred to as a forced choice type. In this study, five choices were given: unsatisfactory, needs improvement, fair, satisfactory and excellent. Students were given the opportunity to respond in a neutral or non neutral way. The areas of focus on the scaled part of the survey dealt with the objectives for the assignment, the instructions provided for students to get started with the assignment, the instructional method used and students' views of the assignment overall. The open-ended questions asked what students liked and what they thought needed to be improved.

Setting. This pilot study described the learning experience of doctoral nursing students as they practiced consensus building skills through electronic communication. The UNIX computing facilities supporting EMail and the local BBS used for the course were supplied by the Medical ARchival System (MARS) project, developed and administered by the Office of Biomedical Informatics at the University of Pittsburgh. Sixteen doctoral nursing students participated in the study over a seven and a half week time period. This was the time in which the course, "Seminar on Structure of Nursing Knowledge" was offered.

Procedure. At the beginning of the course, students were informed of specific readings dealing with knowledge use and that they would need to know this material several weeks later for an assignment. This early announcement was given to allow some lead time. During the third week of the course, an orientation to EMail and the concept of the BBS were provided in the school computer lab. Students practiced sending each other EMail messages and were given more detailed printed instructions that related consensus building to the readings. Knowing the goal was to build consensus, they would have at least three opportunities, once a week for three weeks, to dialogue with classmates via EMail about the readings. Since most students did not have experience with electronic communication and the course was only seven and a half weeks in duration, it was decided to teach EMail skills and provide an interface by the instructor from EMail to the BBS.

During week one students were asked to look for themes in the readings and to develop a framework among them. Then they would send EMail messages to the BBS that described their position related to what the unifying framework should be. For week two they were asked to read the messages posted on the BBS by their classmates, to look for an ideal framework and to consider areas of similarity and difference. Then they would send a message containing an argument to support more than one of the positions posted on the BBS. For week three, they were asked to read the postings of week two and to support by argument more than one of the positions from the messages posted. During the final week of the course a debriefing session was held to discuss the experience and to administer an evaluative survey.

Assignment Preparation. The assignment included readings previously discussed that analyzed consensus building as a process (Ward, 1983a; and Ward, 1983b). The readings that were to have their content synthesized (Dervin, 1983; Nadler & Bozeman, 1983; Rich, 1983), centered on the use of information and how to achieve this goal. The following briefly summarizes the readings to be synthesized by the students.

Dervin (1983) distinguished the components of observer construct versus user construct in relationship to information. From the observer perspective, information is absolute and unbiased reflecting a positivistic view. From a user perspective, information is flexible and shaped uniquely by each person. People may seek information anywhere when conducting day to day practical problem solving. Synthesized information includes generalizations, identifying gaps in the information, and determining implications of the generalizations for practice. The types of synthesis products to be produced need to be considered given the nature of information seeking and the various ways information may be classified.

Nadler and Bozeman (1983) emphasized the components of planning and design to make information useful. A clear purpose must be established at the beginning that will promote user access to the information. This should to be done during the planning and design work whereby compatibility with other operating systems as well as personal perceptions and attitudes are addressed. A number of types of knowledge synthesis products will better meet the varying information needs of users. A method called a Purpose Design to promote use of information was described. The Purpose Design differs from conventional planning that often lacks how implementation should be done. The Purpose Design works more toward how a specific problematic situation can be solved. It accomodates to changes and encourages continued improvements even after the immediate problem has been resolved.

Rich (1983) focused on use of knowledge related to problems and problem solving. Although problems are rarely solved in their entirety, progress can be made toward solving them. To promote knowledge use, a key question centers on determining the type of knowledge synthesis model that would be sensitive to problem solving styles of people and the constraints that are identified. Traditionally, knowledge synthesis has been viewed as an objective summary of what is known in an area and has not been aimed toward any particular practical agenda. A model aimed toward knowledge synthesis should accomodate short long term information needs of decision makers. Synthesis products recommended included a question and answer format, a summary of main research and a briefing book for information on broader as well as sub issues.

Results. The first week's EMail messages by students consisted primarily of identifying main themes from each of the readings and stating a position to reflect the readings. Examples included: a) "These authors recommend establishing a research-practice link in the beginning." This position emphasized having a purposeful foundation at the start from which to do subsequent applications. b) "Potential users must have input or they will not use the system effectively when it is developed later." Here the position emphasized the importance of knowing from the user what is relevant in order to include relevant features into the eventual information system.

During the second week, students showed attention to the messages from the previous week and attempted to develop unifying positions in their arguments. Examples were more extensive and included: a) "Last week's messages may be summarized by stating that in the three chapters knowledge synthesis and interpretation are presented as the means of filling the gap between research and practice. The language used in the synthesized information must be user oriented, that is, understandable. Unless information is constructed in a language that has meaning in the real world of the user, the information will not be used in practice." This message argued that clear relevant language is needed for the information user in order to be able to link theory and practice. b) "In response to all the thought provoking messages some key points are becoming apparent regarding the use of information: meaningful, easily obtainable, practical and adaptable." This message attempted to assemble main concepts within the previous week's messages thought to be related importantly to use of knowledge. Later a framework, consisting of these concepts, could be constructed

During the third week students' messages showed identification with main ideas. Examples included: a) "We all agree that facilitating the use of knowledge should be a main focus when structuring knowledge. Use of knowledge by the practitioner is not a final end point however, but a beginning. As the structure is used, we learn how to revise it so that it improves." In this message use of "we" shows identification with ideas about developing a knowledge structure to be used and making practical revisions following feedback from users. b) "We are going to have to be much more innovative in the way we get knowledge out to large numbers of people. I think this makes the argument for starting new nurses with increased computer comfort as early as possible. Part of the job of using information today is being able to access it by computer." This message also uses "we" in identifying with the main ideas and connects the importance of use of knowledge with the importance of computer skills.

Evaluation. The evaluation survey was administered during the last class. The majority of ratings were in the satisfactory to excellent range. Students found the objectives for the assignment clear and pertinent. The instructions enabled them to work toward completion of the assignment. The main suggestion for improvement had to do with time. While students reported an improved comfort level with computers as a result of the experience, they also expressed a need for more time to acquire the computer aspects before actually getting into the assignment. Other favorable comments included: a) the active participation in learning, b) learning how to read for the purpose of synthesizing, and c) working together with other students. Using the EMail and BBS concept to promote consensus building skills did seem to be an appropriate choice for a teaching strategy.

Implications for Nursing Education

Consensus Building. Practicing consensus building gives students an opportunity to sharpen synthesis skills and to consider the viewpoints of others in an organized way. Because reaching a consensus can help to identify and organize current knowledge on a complex problem, research opportunities may be enhanced. In synthesizing, the student examines published research and observes for continuities as well as discontinuities in the research. The student is able to determine priorities of research that need to be conducted. This understanding of priorities supplies a basis for building a broader span of agreement with peers.

Information Intensiveness. Present day trends suggest that the environment in which nurse educators will practice in the future will continue to increase in information intensiveness. Nurse educators must have knowledge of electronic technology to manage information effectively for curricular purposes. Success in managing information for educational goals will come more readily to those who have, than to those who do not have, electronic technology skills. Every opportunity nursing students have to engage in electronic communication will strengthen skills in dealing with masses of information.

Educational Applications. Specific computer courses are important in any educational program preparing nurse educators. Throughout the educational program experience, all students and faculty should use an electronic mail system as a supplement to other forms of communication. Electronic mail systems offer both students and faculty members the ability to send messages of varying lengths with the advantage of quicker response times. This eliminates the frustration and time wasted in repeated attempts to reach a busy person by phone while at the same time helping to further develop each person's EMail communication skills.

Students and faculty should develop electronic database searching skills for literature. Reference librarians will continue to be needed for more exhaustive searches and should be consulted if an initial end-user search is found lacking. Information about how to access a variety of electronic bulletin board systems should be readily available for students and faculty. The BBS supports colleageal relations, offers focused information and expands communication opportunities across regions and the nation.

Long Term Perspective. By using electronic communication, students and faculty develop stronger skills in managing information. It is important to realize that one computer course, or several experiences with electronic technology, will not be enough for a person to maintain electronic expertise. Technology experiences must be applied with a long term perspective throughout a program. The total faculty body must be committed to facilitating electronic technology. Learning and updating will be necessary since electronic technology is always changing. It is important for students to realize that they will lose any skills they have developed in their school program if they do not continue to work with computers and electronic communication. This learning-updating cycle will provide an excellent forum for faculty and students to collaborate and share their electronic technology expertise.

Implications for Nursing Research

Pilot Research. Small scale pilot studies can provide baseline data for electronic applications in nursing education. They can help to show the way for the development of larger research that is more comprehensive and perhaps even involves a number of different settings. Emerging technologies create immediate effects as well as later effects (Liu, Kolodny & Stymne, 1990). Immediate effects often involve the contents of work and the skills involved to accomplish the needed work of an institution. Later effects could involve sweeping changes in an entire institution or organization. Because so much is still unknown about the effectiveness of electronic communication in the work of teaching, this is an important area for promoting research.

Collaborative Research. Combining the knowledge of a number of people, sometimes from different disciplines, may be needed to conduct an instructional research project that incorporates technology. To introduce this consensus building strategy using EMail and the BBS for example, the authors pooled their knowledge and skills. The local bulletin board for the class was initiated by a technical computer expert who was consulted from the MARS (Medical ARchival System) project at the university. Without this type of collaboration, complex technology based research would not be possible.

Summary

The purpose of this pilot study was to gather beginning information about doctoral nursing students' views in using electronic communication to build a consensus. EMail and a local BBS were employed as vehicles to promote consensus building skills. Incorporating content on the use of information along with a synthesizing process, the strategy was implemented into a course that dealt with knowledge structuring. During the last class an evaluative survey was administered focusing on the students' satisfaction regarding the goals of the assignment, the instructions and the electronic format. The majority of positive comments support continued use of EMail and a BBS to strengthen consensus building skills. Based on survey feedback, increased time to learn the computer aspect of the assignment will need to be considered for classes in the future. Electronic technology must be incorporated as a teaching strategy in graduate programs. To deal successfully with the rapidly growing amount of information important to professional practice, increased electronic communication skills are and will continue to be essential.

References

Burns, N. & Grove, S.K. (1987). *The practice of nursing research: Conduct, critique and utilization.* Philadelphia, PA: Saunders.

Dervin, B. (1983). Information as a user construct: The relevance of perceived information needs to synthesis and interpretation. In S.A. Ward & L.J. Reed (Eds.). *Knowledge structure and use: Implications for synthesis and interpretation.* (pp.153-183) Philadelphia, PA: Temple University Press.

Dvorak, J.C. & Anis, N. (1990). *Dvorak's guide to PC telecommunications.* Berkley, CA: Osborn McGraw-Hill.

Hudgings, C. (1990). Library on-line. *Reflections, 16*(2), 19.

Liu, M., Denis, H., Kolodny, H. & Stymne, B. (1990). Organization design for technological change. *Human Relations, 43*(1), 7-22.

Nadler, G. & Bozeman, W.C. (1983). The relationship of planning and knowledge synthesis. In S.A. Ward & L.J. Reed (Eds.). *Knowledge structure and use: Implications for synthesis and interpretation.* (pp. 515-549) Philadelphia, PA: Temple University Press.

Phillips, G.M. & Santoro, G.M.(1989). Teaching group discussion via computer-mediated communication. *Communication Education, 38* (4), 151-161.

Polit, D.F. & Hungler, B.P. (1983). *Nursing research: Principles and methods* (2nd ed.), Philadelphia, PA: Lippincott.

Rich, R.F. (1983). Knowledge synthesis and problem solving. In S.A. Ward & L.J. Reed (Eds.). *Knowledge structure and use: Implications for synthesis and interpretation.* (pp. 285-312). Philadelphia, PA: Temple University Press.

Roberts, C.A. & Burke, S.O. (1989). *Nursing research: A quantitative and qualitative approach.* Boston: Jones and Bartlett.

Skaggs, B. (1990). Electronic networking. *Communication How-tos, 5*(1), 1-2.

Staggers, N. (1989). Electronic mail basics. *Journal of Nursing Administration, 19*(10), 31-35.

Ward, S.A. (1983a). Documentation of a consensus building approach to knowledge synthesis. In S.A. Ward & L.J. Reed (Eds.). *Knowledge structure and use: Implications for synthesis and interpretation.* (pp. 551-574) Philadelphia, PA: Temple University Press.

Ward, S.A. (1983b). Knowledge structure and knowledge synthesis. In S.A. Ward & L.J. Reed (Eds.). *Knowledge structure and use: Implications for synthesis and interpretation,* (pp. 19-44) Philadelphia, PA: Temple University Press.

Wigand, R.T. (1988). Integrated services digital networks: Concepts, policies, and emerging issues. *Journal of Communication, 38*(1), 29-49.

Wilson, H.K. (1989). *Research in nursing* (2nd ed.). Reading, MA: Addison -Wesley.

Appendix
Consensus Building Through Email
An Evaluative Survey

This survey seeks to determine the effectiveness of the EMail assignment for providing you an opportunity to synthesize knowledge through a consensus building approach. You are encouraged to be frank in your comments; any positive comments and where you see a need for improvement. In this way the assignment can be refined for future use. For the following items please circle the number which best describes your judgment: **1.** unsatisfactory, **2.** needs improvement, **3.** fair, **4.** satisfactory, **5.** excellent. When you have completed the survey return it to the instructor. Thank you.

Assignment Components **Comments**

1. Unsatisfactory 2. Needs improvement 3. Fair 4. Satisfactory 5. Excellent

Assignment Components		Comments
Objectives		
1. Objectives were clear.	1 2 3 4 5	____________
2. Objectives were pertinent.	1 2 3 4 5	____________
Instructions		
3. Instructions were short without sacrificing clarity.	1 2 3 4 5	____________
4. Instructions were logically developed.	1 2 3 4 5	____________
5. Instructions enabled me to complete the assignment.	1 2 3 4 5	____________
Instructional Method		
6. Content of readings fit the stated objectives.	1 2 3 4 5	____________
7. I obtained greater understanding of the readings as a result of the instructional method.	1 2 3 4 5	____________
8. Obvious opportunities existed to practice synthesizing knowledge.	1 2 3 4 5	____________
9. Active participation in the learning process was facilitated.	1 2 3 4 5	____________
10. EMail was an appropriate vehicle to communicate synthesis of ideas to peers.	1 2 3 4 5	____________
11. EMail gave a convenience regarding my time.	1 2 3 4 5	____________
12. Comparing the impact of communicating by EMail with face to face communication offered insights.	1 2 3 4 5	____________
13. Having an EMail orientation in advance assisted my ability to work through the assignment.	1 2 3 4 5	____________

1. Unsatisfactory 2. Needs improvement 3. Fair 4. Satisfactory 5. Excellent

Overall Assignment

14. The assignment complemented other course requirements. **1 2 3 4 5** ____________________

15. The assignment stimulated me to engage in knowledge structuring. **1 2 3 4 5** ____________________

16. The assignment increased my ability in consensus building. **1 2 3 4 5** ____________________

17. The assignment assisted me to learn more on an independent basis. **1 2 3 4 5** ____________________

18. The assignment is relevant to professional nursing today. **1 2 3 4 5** ____________________

19. The assignment will help me to work electronically in the future. **1 2 3 4 5** ____________________

20. I enjoyed this assignment. **1 2 3 4 5** ____________________

Overall Impressions

21. What did you particularly like about the assignment?

22. What improvements need to be made?

23. Please indicate any additional comments or questions here. Thank you.

Projects
Video Communications

Student Communications through Multimedia Technology

John Little
Palo Verde Middle School

Sandee Holiday,
Royal Palm Middle School
Phoenix, Arizona

Abstract

Palo Verde and Royal Palm Middle Schools students are using multimedia technology as means to communicate to other students, to staff members, and to parents. The project provides motivational and educational student communication activities. The project presents practical applications in multimedia and communications technologies. The students are involved in the following three area of communication projects:

1. Students produce a video announcement tape that is shown daily over the Channel One Network video system to each classroom. Students use several computer graphic programs, digitizer, video camera, audio tape player, and VCR to produce the tape

2. Students produce signs, banners, and certificates upon request from teachers and administrators, using computer graphic programs. These are used in the classroom and posted around school to improve communications or provide positive recognition of student achievement.

3. Students produce a school newspaper using word processing, desktop publishing, digitized computer graphics, and computer graphics programs.

These activities have provided the opportunities for middle school students to use multimedia technology as a creative tool and for problem solving. The activities are great for teams of students working together to practice the strategy of cooperative learning.

Restructured Teaching For Distance Education

Don Knezek
San Antonio, Texas

Abstract

Current literature relating both to restructuring and to the teaching profession is rich with visions of teaching roles reshaped for the future. Examples of implementations of these visions, however, are scarce. At Region 20 Education Service Center in San Antonio, high school instructors of foreign languages, mathematics, science, and social sciences are using a technology-intensive teaching model and are teaming with technical, clerical, paraprofessional, and professional staff to effectively teach large numbers of students.

Related to information technologies, the model takes advantage of one-way video and text transmission via both satellite and cable, two-way audio communications, voice mail, sophisiticated video production, laser and scanner technologies, and traditional computing applications. Instruction is live and interactive, and instructors maintain non-standard teaching and office schedules. In addition to the lead teacher, the instructional team is made up of onsite adult facilitators, professional-level teaching assistants and contracted student evaluators, paraprofessionals for assisting with a variety of tasks, and a full television production crew.

This session characterizes the Region 20 instructional model and frames it in current literature on professionalization of teaching. In addition, results of internal and external evaluations are reported. Plans for modifying the model for the 1991-1992 school year, and the rationale for those modifications, are outlined.

Making Tremendous Videos

Patty Achey-Cutts
Cedar Falls, Iowa

Abstract

As the restructuring of schools and preparing students for the twenty-first century are examined, several key issues keep reappearing. Issues such as: student involvement, teacher as a facilitator, guide, mentor, cooperative learning, interdisciplinary approach to teaching, need for communication skills, higher order thinking skills, and authentic projects are some of the main concerns educators are challenged with as they discuss improving student learning.

Making Tremendous Videos or using multimedia with students is a significant opportunity to impact student learning. Students working in cooperative groups on interdisciplinary projects, create and produce multimedia presentations from information they gather, analyze, and interpret.

To accomplish multimedia presentation,s students evolve through a variety of experiences that build their confidence and skill in using the equipment. Once students know how to use the equipment and operational skills become automated, they begin to increase their capacity to think with these media tools. Camcorders, computers, electronic slide show programs, image enhancing programs, tape recorders, and boom-boxes all become resources that enable students to make new connections and associations. The pedagogy of multimedia encourages the development of learning networks and allows the teacher to assume the role of a facilitator and coach.

Projects
Doing the "Write" Thing

Talking Computers and At-Risk Students

Dorothy Noyce, Cathy Conger
McKinley Elementary School
Iola, Kansas

Abstract

Educators have become concerned with the lack of writing skills of today's young peple. Word processing programs offer an exciting medium for writers of all ages. The addition of speech synthesis to word processing has made the activity possible for beginning and remedial students. Students write longer passages on the computer than on paper and tend to edit more. The added modality of listening gives writers an opportunity to monitor their work and develop a sense of audience. The robotic voice of the computer serves as a source of amusement and motivation.

Cross age tutoring with older students using the Language Experience approach is used to help both beginning writers and remedial students to improve their communication skills. Peer editors work with authors at each grade level at the computers. Resulting individual and class books are bound for circulation. supporting a strong writing emphasis in the school are such activities as Sustained Silent Writing, attending Young Authors Conferences, Writing Across the Curriculum, Cooperative Learning, and opportunities to circulate student's work

The presenters will discuss talking software and idieas for turning both students and teachers on to various uses of computers. Staff development ideas will be presented. Grant writing will be mentioned. The presenters received a 1989-1990 Educational Enhancement Grant from the State of Kansas.

Do the Write Thing - An Innovative Computer Writing Project

John Scheiss
Rochester, New York

Abstract

Project "Do the Write Thing" is an elementary school computer writing project funded by the Federal Fund for Innovation Grant program. It began at four elementary school sites in the Rochester New York City School District in the 1989 - 1990 school year as a result of a grant written by two teachers, Terence Sweeney and John Schiess.

"Do the Write Thing" was designed to help increase the writing skills and *motivation to write* of elementary students who were considered "at risk." The program emphasizes simultaneous instruction about computer use and the writing process, publishing of student work and sharing that work with a distant audience. Existing hardware and software supplied by the school district was supplemented by Macintosh computers, publishing software, high quality printers, and book binding materials to enhance the publishing effort.

Students in the project participate in book and newsletter publishing, after-school writing clubs, and will begin mentoring with professional writers. Other schools and districts are involved through the use of telecommunications. Communication (shared writing activities) among local area sites is also a major focus.

Results have been encouraging. Anecdotal information collected from students shows a greatly improved attitude towards writing. The project has been expanded this school year to include five more sites.

Promoting Writing With Primary Students in the Computer Lab

Patricia Friesen
Lincoln Public Schools, Nebraska

Abstract

The "First Grade Writing Lab" is a joint effort by the classroom and computer lab teachers to foster writing and technology connections. This session will focus on the capabilities of primary students.

The context. The computer teacher has an Apple IIe for each student booted up, laminated paper keyboards (8 1/2" x 14"), printers, a reading corner with backjacks, and two tables for drawing and writing.

Logistics. Students work individually and with partners. The lab instructor, to maintain a consistent environment with the classroom, encourages students to sound out for themselves any word they want to write, and asks students to read back what they are writing. Initially, the lab instructor had all computers booted with the student's previous day's work before they arrived. When they left, the lab instructor saved all writing on individual student disks.

The computer's word processing capabilities play a significant role in developing literacy skills. Locating a letter on the keyboard promotes initial letter discrimination. Coordinating upper and lower case, using ending punctuation, and quotation marks follow. The word processor is a motivating medium that allows students to publish their own writing as books in enlarged *LogoWriter* font.

Paper

Computer Mediated Conferences for Collaborative Learning: Towards Hypermedium?

*A.Derycke *, D.Clément, C.Ladesou, C.Viéville **
Laboratoire TRIGONE, Institut CUEEP - Université de Lille I
France

Abstract :

The use of Computer Mediated Conferences (CMC) has gained popularity in the various fields of human activities. Some pioneers have reported experiments in the field of education either to support remote learners in Open Learning institutions, or to teach at distance. We have designed a first generation of CMC tailored for education, and based on the French Videotex network. We have also tested this and the synthesis of the first results are reported here. The human communication process, even if it is mediated by a computer, is very complex and requires the contribution of a lot of disciplines such as psychology, linguistic or social sciences. The design of a new generation of CMC adapted to education goals, is an ambitious task which calls for very flexible work benches to allow for multiple and various experiments. It also requires the means to solve any number of fundamental issues, of which seven are highlighted, and to explore new foundations and paradigms for such a design.

Introduction

The CUEEP, an Institute of the University of Lille, is involved in further education and in research and development in educational engineering. It has much experience in the use of Computer Assisted Learning (CAL) and in the design of multimedia packages for Open Learning. For two years there has been a major shift towards a global architecture of an Open Learning system based on the advances in new communication technologies.

That is done by the setting up of networks either at the level of the classroom and learning resource centers (Local Area Networks) or at a regional level. Previous experience, especially with LAN [Derycke 87], has convinced us of the need to find a balance between the necessary individualization of learning (self-paced, plurality of cognitive styles) and the necessity of a socialization of learners, especially those who are distance learners, and a promotion of cooperation. We have called this concept "Cooperative autonomy" [Derycke 87].

The second generation of open learning systems or Open Universities is characterized by two things : the first is the "mediatization" of the curriculum via various material (paper, TV, audio...) and the second is the asynchronous delivery of the contents of courses. In spite of its success the second generation has severe drawbacks especially in respect of the interactivity of the learners with the media and their isolation. Of course, the use of courseware can reduce the first drawback, and teletutoring by telephone or regroupment periods can decrease the impact of the second one.

But we think that to have a larger vision which takes into account, not only the advances in new communication technologies, but also new theories of education, is necessary to set up a third generation Open University.

For this purpose, it is important to conduct both theoretical and experimental research. In this paper we will emphasize one aspect of our work devoted to the use of multimedia computer conferencing systems in distance education. From our first experiment with the CMC we have designed, we can highlight seven of the major issues that must be addressed by the next generation of CMC dedicated to education.

I. Computer Conferencing and distance education : the CoCoNut project

I.1. The computer conferencing system

The computer (tele) conferencing systems (or computer mediated conferencing) have gained more popularity these last few years [Byte 85]. There are asynchronous communication systems or connectionless communications : the actors (i.e., learners, tutors, teachers) do not interact with the network at the same time. The messages, usually textual, are stored on a conference server. These are different from electronic mail because they are more group-oriented. The first use of such systems was in the field of cooperation between scientists or managers to help them discuss about topics, problems ....

Now, a large public uses CMC to exchange information in various domains of interest, from computer programming to astrology; in this case they are called Bulletin Board Systems.

Hiltz [Hiltz 78] has shown the importance of computer conferencing systems in education and she has stated that "it was recognized by those who followed this path (introducing

---

* and Laboratoire d'Informatique Fondamentale de Lille (LIFL)

CMC), that communication was an important part of the educational process; communication not just between teacher and student, but involving peer-group reinforcement among students as well." In spite of this premonitory view, the CMC has only recently received interest from the educational world. This is due on one hand to the cost of hardware and the availability of computer networks, and on the other hand to the difficulty of finding a place for these tools in the educational process, even in distance teaching.

Some pioneers have conducted experiments. Haile, for example, [Haile 84] discovered that CMC was qualitatively different from an electronic mail system and can provide an educational environment which fosters group-oriented communication and greater interactivity.

1.2. The design of a CMC supported by Videotex network

For two years we have had launched an ambitious multi-year, multi-disciplinary research project call CoCoNut, for Cooperation and Communication Network Utilities for teleTutoring. The project is related to the study of various communication tools for the setting up of a third generation Open University.

Beside theoretical research work, some prototypes are designed and tested. There are two important sub-projects:

- the first one deals with small group multimedia real-time interaction which is some sort of computer assisted audio conferencing or electronic blackboard [Derycke 90]
- the second project deals with asynchronous communication with a larger audience. We are going to present this second project.

We have chosen to design a Computer Conferencing System from scratch for three reasons:

- first, even if good systems existed, like Cosy or Party [Byte 85], they were not available to us when we started our project. Our first evaluations showed us that these CMC were relatively complex for the unexperienced user and, of course, that their interface was in English and not in French.
- we wanted to take advantage of the success of the French videotex network. There are more than 400 000 Minitel terminals in the homes of our Region. So, for part of our experiments, it is not necessary to provide terminals to the learners. Other advantages are brought by the videotex network such as equal access and same cost on all the territory or a relatively well known user-interface (special keys of the terminal called Minitel).
- The system must be a platform for various experiments so it is important to master the software sources and to have the possibility of customizing the conference or having incremental changes in the underlying system.

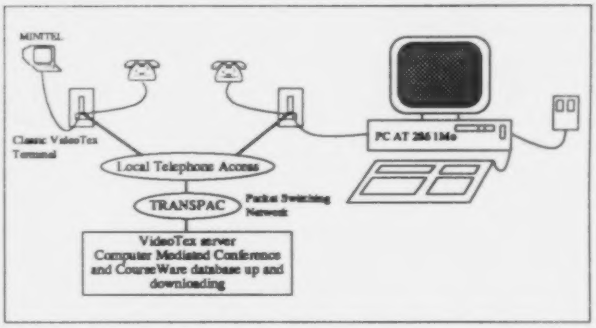

MINITEL
Classic VideoTex Terminal
Local Telephone Access
TRANSPAC
Packet Switching Network
VideoTex server
Computer Mediated Conference
and CourseWare database up and downloading
PC AT 286 1Mo

Figure 1. General architecture of the CMC network system)

Figure (1) gives a general overview of the architecture of our system. The videotex server is based on a set of Unix minicomputers linked by an Ethernet LAN. The CMC is implemented in C language and is built on top of a videotex software kernel. This is viewed as another service which is accessible in the same way as other videotex services (information, systems, traditional electronic mail, text retrieval system...). The CMC is coupled with a database and some procedures to up-load and down-load data other than text ones. As a matter of fact, the CMC is accessible either from a minitel terminal or from a PC microcomputer. It the latter case, the PC emulates the Minitel but it also offers other possibilities such as exchange of data, computer programs, images....

The user interface was kept simple, only a few commands and a syntax which are very similar to the guidelines of the French videotex. All the documentation for the casual users is resumed in a six page booklet.

Of course, there are other functions which are available for the experienced users (teacher, tutor, some learners) or for the CMC administrator, but they are not available to the other users.

For the presentation of the structure of the various conferences or sub-conferences we have used the metaphor of the electronic campus : conferences take place in different rooms where it is possible to discuss specialized topics with other people. We have also provided some rooms to allow informal conversation : the "café".

II. Experimentation of the CMC in Open Learning

We have conducted several experiments with our system in the general framework of the development of Open Learning in our institute : to give more choice or freedom of the time, the rhythm, the place. Distance teaching/learning is only one mode of education we are developing amongst others.

But before we can draw several results from our experiments it is necessary to give some basis of our analysis.

In fact communication between humans is very complex and can carry of lot of different interpretations or points of view.

II.1 Some human factors for the analysis of the experiments

There are different levels of analysis of the process of communication via a CMC. In reality, these levels are very intricate and correspond to different human factors with regard to the speciality or background of the observer.

We can roughly separate different types of human factors into four groups:

- Ethical, political or sociological factors : is this use of CMC compatible with the development of democracy ? More generally, how does the use of new communication technology fit in with the political project of a new generation of Open University, of open learning ? Other problems are with respect to privacy, legacy....
- Social psychological factors which deal with the organization of groups, the interaction between their members, place and value of etiquette, evolution of the meta-communication, codes and conventions.... From our pedagogical point of view emphasis was put on the social construction of knowledge and the necessity of cooperation and emulation [Doise 84].
- Cognitive psychological factors deal with the ability of the users to interact with the CMC system to accomplish different tasks in relation to the general goal. This is also, in the pedagogical field, the development of transverse skills with regard to access to information and the development of a plurality of learning styles.
- Sociolinguistic or conversationalist factors which are related to discourse analysis, the "turn-taking" aspect, the Speech Act Theory [Austin 62] [Searle 69], ...

All these human factors have already been of interest in some analyses of the CMC experiment, see for example [Bellman 89].

Our project and observations are transverse to these factors because we want to have a global view which puts emphasis on the total education process.

It is important to note that these human factors are important not only to provide us information on the experiments but also to modify the existing CMC or to give a basis for the design of a new CMC.

II.2 A first analysis of our experiments

II.2.1 Conditions of experimentation

Four different groups of learners at undergraduate level have tested our CMC. In this first stage all access was through a videotex terminal. Nobody reported difficulty in access to the terminal (= Minitel). For two of these experimental groups CMC have only a secondary place. Most of the educational process still remains traditional : multimedia package delivery, distance tutoring by telephone, different periods of meeting in learning resource centers where the learners have the possibility to meet other learners on the same curriculum. There are informal assignments given on the use of CMC. The presentation of the possibilities of CMC emphasized on conviviality and free use.

The two other experimental groups are of more interest for us, because we have chosen to give a more important, central place to the CMC in the learning process. Learners received a formal assignment which obliged them to use the CMC. Their interaction with the system was part of the process of graduation. Two types of activities were chosen :

- a group problem solving activity. As a matter of fact, it is recognized that this kind of pedagogical activity is a promising potential of CMC in education [Beckwith 87]. The activity is organized around a collective mathematical game (Math-Hangman). The moderator of this conference has a lot of experience in the management of a traditional classroom with this collective game. It must be noted that in this game all each propositions given by a learner must be submitted to the group and that it is applied to the problem only if it receives a majority agreement.
- a case-study activity : this activity takes place in an economics course of the company included in a larger curriculum for the training of office automation system managers. The case-study is used as a synthesis of the course. The assignment is a study of the investment in office automation in Small or Medium sized Companies. The moderator gives the first questions and the general direction. The activity implies that the group designs a collective set of questions for an investigation in the S.M.C. He chooses the panels of representatives S.M.C. Each learner has one or two S.M.E to interview and he reports the results of the investigation using the CMC. The results are collectively discussed (validity, generalization,...) and a collective report is written. This approach is close to those of the project pedagogy envisaged as a special form of distance pedagogy [Dirckinck 90].

The materials, which are available to analyze experiments are:

- the report of the tutor/moderator : observation during the briefing and de briefing sessions in traditional classroom meetings, accreditation results, ...
- the history of all the messages exchanged during the CMC sessions (one, two, or three months),
- non-directive interviews of some learners.

II.2.2 A synthesis of our first results

The first analysis is related to the usage of our CMC system. We have not encountered real difficulties. In fact practically all the learners (aged from 19 to 30 years) had already used the Videotex system : terminal and access procedures. They reported no difficulty with the learning of the CMC system. Fortunately the complaints were directed at the general

Videotex system : lack of professional keyboard on the terminal, small screen with only 40 columns, slowness.... But these drawbacks are well known to learners, and thanks to the French videotex success everyone accepts that...

At the social psychological level, all our investigations confirm the previous experiments: [Bellman 89]:

- the lack of meta-communication, even if new textual conventions can be found, they are seem to be difficult at the first time of usage. But after a first step this difficulty seems less important and most of the textual conventions disappear.
- the major problem which appeared from our experiment is due to the multi-thread of the discourse. This is the result of the asynchronous writing of messages in the CMC. In this first version of our CMC there is only a time stamped, on the messages when they are created and a sequential number which indicates the disposal order in the CMC server. Because it is difficult to have a contextual view of a particular message and because tools to give a graphical historical view of the conference with different links between messages do not exist, learners are sometime lost. Some messages request "could someone tell me what we are speaking about ?. At this time the only solution to this problem is the action of the moderator which periodically tries to resume a sequence of messages.

For us the asynchrony is probably the most important characteristic which we will take into account in the design of new CMC systems. This has been recognized by major specialists of the CMC field or by the man-machine interface one [Bowers 88] [Norman 86].

- At the linguistic or conversation level it appears that roughly 50 percent of the messages are not really related to a conversation process [Bowers 88]. As a matter of fact, it is difficult to attach these messages to a particular "turn taking" of the conversation. However there are some sequences of messages which can be really analyzed as conversation : in general it implies only two locutors and the time between the different messages tends to be shorter than average. In fact, there is a shift towards a more real time interaction. It can also be noted that, in that case, the engagement of the locutor seems to be higher.
- At the pedagogical level, two important characteristics have been observed especially in the two last groups. The first one is relative to meta-cognitive information given by some learners. They reveal difficulties in encountering their learning and in criticizing the pedagogical values and methods used in the teaching of the course (especially about papers or reference books). But at this stage, it is difficult to use this information for tutoring.

The second one is the difficulty in organizing the cooperation between learners. The main goal is buried in the "noise" : messages about the organization, cooperation and co-ordination of points of view which mixed up with other ones relative to social activities. The operating sub-conferences is not a good solution in this case. As a matter of fact, we found there is a lack of structuring tools which would enforce structuration and cooperation. The free form of the textual messages is not well adapted to these goals.

With the first generation of CMC the only way to overcome these difficulties is to create a precise and well-accepted protocol between the users (this can be explained during a meeting at the beginning of the conference) and to give an important role to the moderator (tutor).

There is a tension between the free (unconstrained) access to the CMC system which would encourage participation and the necessity of structuring (constrained) the activities.

One way is perhaps to design new communication tools by tailoring the CMC system for this type of application. This is similar to the approach of microcomputer systems : between the general programming language which gives the freedom of implementing anything to the programmer and the specialized applications, there is a place for generic tools like spreadsheets.

III. Seven issues for the next generation of CMC system dedicated to education

III.1 The theoretical issue

It is important to have a strong background in theory to give a solid foundation to the design of new CMC systems. These theories are related to the different levels described in the II.1 section. At the philosophical or political level it seems that new approaches like Contextualism [Whiteside 88] could give a methodological framework to reform group meetings and collaborative work. The evolution of general theories of education must also be taken into account. We are especially interested by post-modernism analysis of pedagogy, [Giroux 88] because it reflects our assumption that a balance between individualization and socialization, i.e. acceptance of the difference, is a democratic requirement.

At the linguistic or conversational level it appears that the Speech Act Theory and its extensions could be a basis for the design of collaborative systems which favour participation and co-ordination [Winograd 86].

III.2 The multimedia issue

One of the most important deficiencies of our CMC is its textual orientation with two severe drawbacks : no verbal cues as are used in traditional teaching and its related psychological problems ; and it is difficult to use this kind of CMC in the field of sciences for example, because of the lack of pictures, complex drawings and the possibility of exploiting them.

This can be partially overcome with a use of tools to download and up-load data other than textual messages. This is only possible with a microcomputer connected as a terminal. But a major problem still remains with the integration of the different media especially with the voice.

Voice annotation is still at a research phase. In the real time mode of interaction it is easier to use voices, texts and pictures simultaneously [Derycke 90].

The emergence of new powerful networks like ISDN will surely facilitate the task at the transport level.

III.3 The navigation issue

This is an issue related to the multi-thread of the discourse problems and the disorientation of some users. It is a problem well known to the Hypertext/Hypermedia (HT/M) specialists (for a good introduction see [Nielsen 90]). There is a need for special tools to help user navigation. It the HT/M field it is called a browser. The most important requirement for a new kind of CMC would be the availability of a historical browser which could give not only a chronologic view of the stored messages but also the context of the writer of a particular message. This problem is close to one encountered in the field of collaborative writing of texts and the management of different versions. Research in the HT/M domains will probably give the first elements of solutions [Halasz 88]. Another problem with querying messages is the problem of information overload. It is important to design filtering access which selects messages not only by dates, name of senders but also by the categories (types) of messages. It is also possible to use some active messages, i.e. messages with a relatively autonomous behavior, which are able to collect, for example votes, acknowledgments.

III.4 The group structuring issue

For the design of new CMC systems dedicated to, or tailored for education it is important to make it easy for the moderator (tutor, but also to the learners themselves), to organize and structure groups in relation to the pedagogical goals and the social psychologic characteristics of the target audience [Bellman 89]. The project COSMOS, orientated towards Office Automation use of CMC, proposed a special system language for this purpose. If it is a good solution for professional users who personalized their own environment, it seems difficult to apply in the field of education which is characterized by occasional users. Perhaps visual programming will be a solution.

III.5 The textual conversation issue

At the beginning the possibility of having a dialogue, by textual messages can appear to be strange. The linguistic or Speech Act Theorists have shown that message-text is an utterance of a conversation, so that CMC can be analyzed in the light of conversation theories.

However, the different asynchronies introduced by the storage of messages and asynchronous access introduce some important variances with a traditional speaking conversation. One solution was probably to find system interface on the Conversations For Action paradigm [Winograd 86], [Bowers 88].

In the education field the challenge is to design a system which favours interactive participation without hierarchical and incentive management such as at companies.

III.6 The specialization issue

If we want to support the learning process efficiently with a CMC system it is important to give more tools to support the activity of the learner in relation to the pedagogical goal. It is not possible to have an integrated tool which is tailored for all kinds of activities : such as argumentation, case study, brain-storming....

We do not think that Artificial Intelligence and expert systems could bring solutions to these high level human activities. The tools must only be considered just as systems to make the human intelligence argue. Some developments have been made in this way with some tools to support argumentation, for example the Euclide systems [Smolenski 88] or to support brain-storming decisions like the Colab project [Stefik 87].

III.7 The integration issue: towards hypermedium

From the overview of the first six issues it appears that a CMC tailored to education could be very complex to design. There is a danger of setting up a system which is composed like a puzzle of various tools. The challenge is to design a new system with a homogeneous interface where the communication act is central. The use of hypermedia and courseware must be integrated in this communication environment. A lot of research work must be carried out to determine the best approach for the man-computer interface of such a system. It is important that this integrated environment is underlain by only one or two powerful metaphors such as the desktop metaphor in the world of office automation.

The integration could be based on the concept of HyperMedium described by Gaines "The hypermedium concept is significant in capturing the computer-supported, open-ended access to a diversity of media that offers many possibilities for creative thinking, problem solving and networking by individuals and dispersed groups" [Gaines 88]. This is close to our aims for the design of a third generation Open University.

To reach the integration goal, it is important to have a clear view of the interaction between users and the communication system.

For this purpose, a layered approach can be used to clarify this complex task. The layered model proposed by the E.E.C program RACE [RACE 90] is very useful as a starting point of analysis and requirements for designing a new communication system (figure (2)). The users have goals, purposes or intentions concerning a particular task employing conceptual models (i.e., ways of thinking about how to reach theirs goals and purposes and how to satisfy their intentions) which reflect their cognitive abilities and physiological characteristics. Each of these "layers" can be mapped into related features in a communication

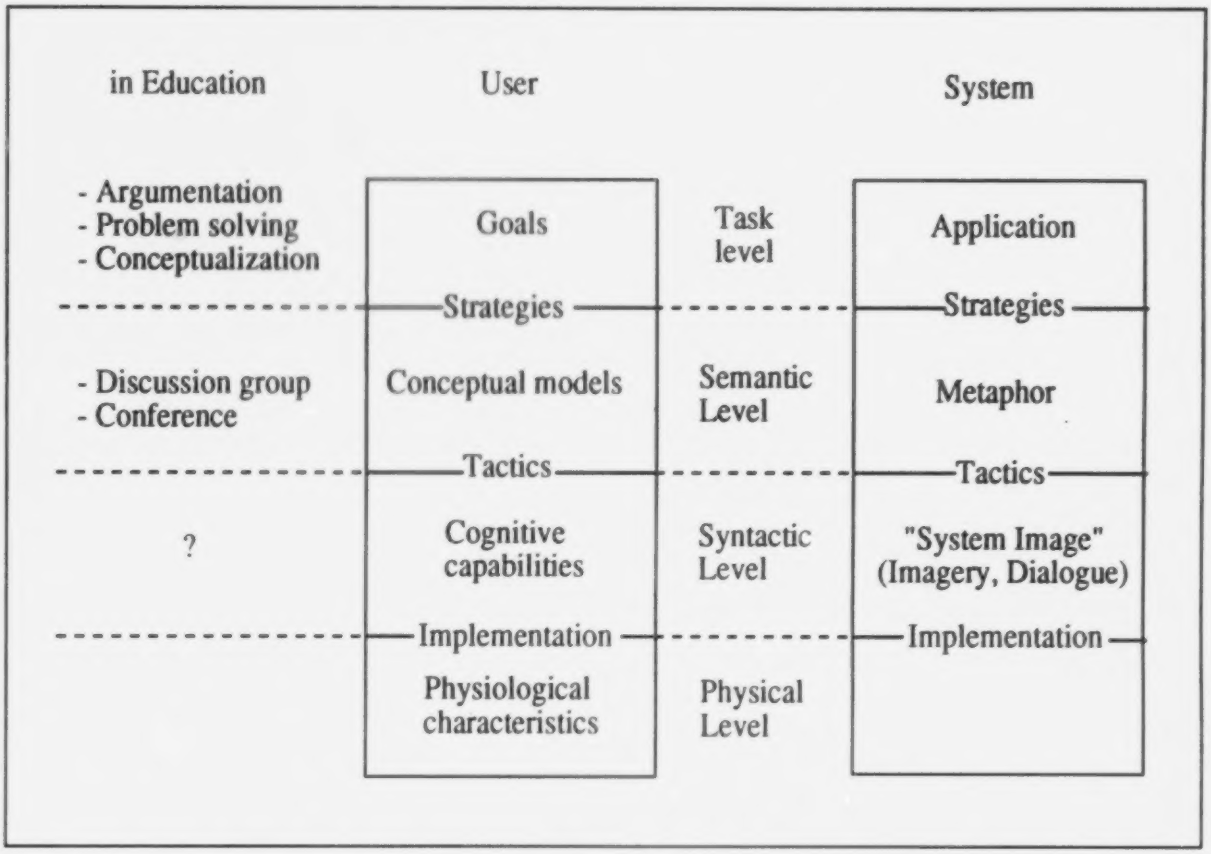

in Education
User
System
- Argumentation
- Problem solving
- Conceptualization
Goals
Task level
Application
Strategies
Strategies
- Discussion group
- Conference
Conceptual models
Semantic Level
Metaphor
Tactics
Tactics
?
Cognitive capabilities
Syntactic Level
"System Image" (Imagery, Dialogue)
Implementation
Implementation
Physiological characteristics
Physical Level

Figure 2: Layered Mapping of user and communication system characteristics derived from RACE 92

system. Metaphors, for example, have to be designed to match the conceptual models held by the target users. The model also implies that a mismatch at any level may prejudice the success of the communication system. The challenge is to design applications which give an active mediation to support pedagogical goals.

At the semantic level the system must present only a few metaphors which have a common "feel" and which are in accordance with the learning activity.

Conclusion

Computer-mediated communications can have a central place in the development of the Open learning hypermedium (i.e., which offers different dimensions to favour further education or lifelong education). However, in spite of the social values of the CMC, like the possibility of introducing noise in a rigid system, it is important to assign a pedagogical role to these tools. For this aim, it is necessary to carry out a lot of field research work and experiments and to design flexible platforms which allow changes in goals, and in human-computer interfaces. The design of new systems must take into account, not only the contribution of cognitive psychology and social sciences, but also some new foundations for design, or new paradigms [Winograd 86].

Bibliography

Austin, J.L. (1962). *How to do things with words*. Clarendon Press.

Beckwith, D. (1987). Group problem-solving via Computer conferencing : the realizable potential. *Canadian Journal of Educational Communication*. spring Vol 16, n° 2, pp 89-106

Bellman, B.L. (1989) *Toward a model for interface Information server design* in Message handling Systems and distributed applications, E.Steffernd, O.J Jacobsen, P.Schiker (ed), North-Holland, pp 427-445

Bowers, J. Churcher, J. (1988) Local and global structuring of computer mediated communication developing linguistic perspectives on CSCQ in Cosmos *Proceedings of the CSCW'88 ACM Conference* - Portland -OR- pp 125-139

Derycke, A. Viéville, C. Vilers, P. (1990) Cooperation and communication in Open Learning : the CoCoNut Project

Proceedings of the WCC'90 IFIP Conference, Sidney, July, in Computer in Education ; A.Mc Dougall and C.Dowling (editors), North-Holland, pp 957-962

Derycke, A. Loosfelt, Ph. Viéville, C. (1987) Towards pedagogy in networks- *Proceedings of the ICCAL Conference*, University of Calgary, May, pp 93-98

Dirckinck-Holmfeld, L. (1990) Project pedagogy and computer mediated communication in distance learning *Proceedings of the WCCE'90 IFIP Conference*, Sidney, July, in Computer in Education, A.Mc Dougall and C.Dowling (editors), North-Holland, pp 963-970

Doise, W. Mugny, G. (1984) *The social development of the intellect*, London, Pergamon Press.

Gaines, B. Vickers, J.N. (1988) Hypermedia design. *Proceedings of the RIAO 88 Conference on User-oriented content-base text and image handling*. M.I.T, 21-24 Mars, pp 14-23

Giroux, H. (1988) Border pedagogy in the age of post-modernism. *Journal of Education*, vol 170, n° 3, pp 162-181

Haile, P.J. Richards, A.J. (1984) Supported the distance learner with computer teleconferencing *15 th annual conference of the Northeastern Educational Research association*, NY, 25 October

Halasz, F.G. (1988) Reflections on Notecards : seven issues for the next generations of hypermedia systems. *Communication of the ACM*, Vol 31, n° 7, July, pp 836-852

Hiltz, S.R. Turoff, M. (1978) *The network nation: human communication via computer*, Viley

Nielsen, J. (1990) *Hypertext and Hypermedia* Academic Press

Norman, D.A. Draper S.W. (1986) *User centered system design* Lawrence Erlbaum Associates - pp 444-445

Race (1990) *Commission of the European Communities.* - operation 92 - Advanced communications-technologies in Europe

Searle, J.R. (1969) *Speech acts* Cambridge - Cambridge University Press.

Smolenski, P. Fox, B. King, R. Lewis, C. (1988) *Computer-aided reasoned discourse or, how to argue with a computer* in "Cognitive sciences and its applications for human-computer interaction" Raymonde Guindon. Lawrence Erbaulm Associates, NJ.

Stefik, M. et al (1987) Beyond the chalkboard : computer support for collaboration and problem solving in meeting. *Communication of the ACM*, January, Vol 30, n° 1, pp 32-47

Whiteside, J. Wixon, D. (1988) Contextualism as a world view for the reformation of meeting *Proceeding of the C.S.C.W'. 88 ACM Conference*, Portland, pp 369-376

Winograd, T. Flores, F. (1986) *Understanding computer and cognition* Ablex, Norwood, NJ

Paper
Composing Process Software and Writing Instruction

W. Michael Reed
West Virginia University

Abstract

This paper reviews four studies in which *Writer's Helper* was used. It was found that the prewriting features of this program are most effective for below-average writers, that many of the revision analyses predict final-product quality and syntactic complexity, in addition to prompting higher-quality types of revision. The final study describes how *Writer's Helper* was integrated in a semester-long writing unit.

Introduction

After over five years of research on the effect of word processing on writing performance and attitude (see Hawisher, 1987, for a listing of studies and their findings), computer-based writing research is being directed to the area of composing process software, computer programs that help student-writers prewrite and revise. This line of research is acknowledging well-researched findings that many writers do not have innate skills for generating, selecting, and organizing ideas—considered prewriting in writing process theory—and skills for revising text they have produced—considered rewriting in writing process theory (Emig, 1971; Flower & Hayes, 1981; Perl, 1979; Sommers, 1980; Stallard, 1974).

When word processing by itself is integrated in writing instruction, it is usually based on the assumption (a) that teachers will instruct and provide non-computer opportunities for their students to develop and employ prewriting and rewriting strategies or (b) that the student-writers bring to the word processing session such skills and employ them as they would during non-computer-based writing tasks. Although the former assumption is a requirement when instructing writers of most abilities, especially poor and average writers, the latter assumption is potentially faulty when instructing medium- and low-ability writers. Research shows that poor and average writers do not have such skills (Flower & Hayes, 1984; Reed & Sherman, 1991). And, although teachers can provide non-computer opportunities to prewrite and rewrite, computers can facilitate the process via their well-recognized benefits: (a) a one-to-one ratio; (b) student-determined pace; (c) motivational; and (d) more immediate and less negative feedback than that provided by teachers, as often perceived by students (Hannafin & Peck, 1988).

Similar to the volume of word processing programs, there are numerous composing process software programs. Such programs include (a) *WANDAH*, (b) *Writer's Workbench*, (c) *SEEN*, (d) *HOMER*, (e) *COMP-LAB Writing Modules*, (f) *Writer's Helper*, and (g) *Wordsworth II* (Wresch, 1984a). According to Wresch (1987), the most-often used composing process program is *Writer's Helper*. *Writer's Helper* is divided into two major components: (a) prewriting and (b) rewriting subprograms. The prewriting component has 20 subprograms grouped according to whether they are used to Find a Topic, Explore a Topic, or Organize a Topic. The rewriting component involves 20 procedures that perform final-product analyses; these procedures are grouped according to whether they are used to address the essay's Structure, its Audience, or Homonyms, Usage, and Gender-Related words (See Wresch [1984b] for a detailed explanation of *Writer's Helper*).

There is some debate over whether it is better to use composing process software such as *Writer's Helper* or to modify existing word processing programs so that writers can receive generative and evaluative assistance as they compose. *Writer's Helper*, along with most of the other composing process software mentioned earlier, is not as true to the recursive nature of writing process theory as are modified versions of word processing programs. With *Writer's Helper* writers must use a prewriting procedure, exit the program and use a word processing program for drafting (in the latest version of *Writer's Helper*, there is a Note Pad feature which allows the writer to draft, but it does not have many word processing features), and then re-enter *Writer's Helper* for the final-product analysis. Some computer-writing theorists/researchers believe that this process isolates the stages of the composing process, violating somewhat the recursive nature of writing. On the other hand, word processing programs modified to provide generative and evaluative assistance when writers are composing are used based on the assumption writers will access them and once receiving help will know how to use that information—although they better accommodate the recursive nature of writing, the ability of the writer will usually determine their effectiveness.

The purpose of this paper is to present four studies in which *Writer's Helper* was central and to comment on the implications the results of these studies may have on computer-based writing instruction. The first study addresses the appropriateness of *Writer's Helper* prewriting procedures for writers of different abilities; the second, how well the revision messages predict essay quality and syntactic complexity; the third, how the revision messages produced different types of revisions and the relationship between these types of revision and quality; and the fourth, how *Writer's Helper* was integrated with different modes of discourse.

Four Studies

Writer's Helper and Prewriting Strategies

As stated previously, writers of varying abilities come to a writing task with different writing strategies: poor and average writers have fewer strategies than better writers. Although lower-ability writers may benefit from computer-supplied prewriting and rewriting prompts, higher-ability writers might be negatively affected by them, mostly due to their simplistic nature.

I (Reed, 1988) assigned 21 prospective English teachers to go through 10 of the prewriting procedures from *Writer's Helper* and identify whether they believed a particular prompt would be most effective for a below-average high school student, for an average high school student, or for an above-average high school student. They also identified for which of the student groups it would be next most effective and then the group for whom it would be least effective. The prospective teachers also provided comments, to explain why they believed a prompt would be most effective with a specific group and least effective with a specific group.

Very consistently these prospective English teachers believed that the prewriting prompts would benefit below-average writers the most with average writers next, followed by above-average (see Table 1). Some inconsistency did occur, however. For example, nine of the English education students felt that below-average writers would benefit the most by the Brainstorms procedure while nine others felt that they would benefit the least. Some of the reasons the former group of nine believed it would be beneficial are (a) it allows them to be free of grammatical concerns; (b) it helps initiate a search of information; and, (c) it helps alleviate writing apprehension because it is easy to use. On the other hand, the reasons the latter group gave for believing they would benefit the least are (a) writers will need more assistance in order to complete the writing task and (b) it imposes a time limit.

Other exceptions to the prompts most benefitting below-average students were the Crazy Contrast and Debating an Issue prewriting prompts. For these two prewriting strategies, average writers were identified as those who would most benefit from Crazy Contrasts because it (a) is an effective way for them to develop new thought patterns and relationships and (b) is sufficiently interesting whereas above-average writers were identified as those who would most benefit from Debating an Issue because it (a) is an effective guide for persuasion and (b) it is challenging.

Fairly consistently below-average writers were identified as the ones who would most benefit from the majority of the prewriting prompts because (a) the prompts are interesting; (b) they help with generating ideas, an area of weakness for such writers; (c) they keep them on task; (d) they motivate them; and (e) many of them help these students with organization, another

Table 1
Rankings of Writer's Helper *Prewriting Procedures Based on Writing Ability*

	Below Average			**Average**			**Above Average**		
Rankings —>>	1	2	3	1	2	3	1	2	3
Finding a Topic									
Brainstorms	9	3	9	4	16	1	7	3	11
Lists	8	4	9	6	15	0	8	2	11
Questioner	12	6	3	7	14	0	2	1	18
Exploring a Topic									
Crazy Contrasts	4	11	6	11	8	2	6	2	13
Teacher's Ques.	8	7	6	9	12	1	4	2	14
Three Ways	10	6	5	8	12	1	3	3	15
Organizing Information									
Trees	12	4	5	5	16	0	4	1	16
Debating Issue	7	4	10	6	15	0	8	2	11
Compare/Contr	12	3	6	4	17	0	5	1	15
Five-Paragraph	12	5	4	7	14	0	2	2	17

area of weakness for this ability student. Also fairly consistently above-average writers were identified as those who would least benefit from the majority of the prompts because (a) they already have the target-skills and (b) the prompts are too simplistic and they might become bored with them.

This study addresses prospective teachers' perceptions of the appropriateness of the 10 prompts. Two other steps are suggested: (a) have practicing teachers review them and make recommendations and (b) ask students to use them all and identify which ones they liked the most or seemed the most useful. Because often what students like is not always what works best for them, a possible third step is to assign students to different prompts and assess the final writing product for effectiveness. But, the issue might be the difference between the affect and the cognitive—and the goal of the instruction.

Writer's Helper Final-Product Analyses as Predictors of Quality and Complexity

I (Reed, 1989) conducted numerous final-product analyses on 27 essays—nine narrative, nine descriptive, and nine persuasive written by low, average, and above-average college freshmen—from an earlier study (Reed, Burton, & Kelly, 1985). These essays had been holistically evaluated and analyzed on the three measures of syntactic complexity (words per T-unit, words per clause, and clauses per T-unit) in the earlier study. (A T-unit is an independent clause plus any related dependent clauses and embedded modifiers.) I then regressed quality and syntactic complexity scores on the final-product messages given by the computer to these 27 essays. I found that (a) the words-per-paragraph messages were reliable predictors of quality (a positive relationship); (b) the words-per-sentence messages were reliable predictors of quality (a positive relationship) and words per T-unit (a negative relationship); and (c) the readability-index messages were reliable predictors of words per T-unit, words per clause, and clauses per T-unit (all negative relationships). Essentially my findings indicate that such feedback, if followed, would lead writers to produce higher quality writing and more highly complex syntax.

Writer's Helper Final-Product Analyses and Types of Revisions

In another study (Reed, 1990a) 26 college undergraduates were randomly assigned to one of two groups. Thirteen students were assigned to use their innate revision strategies, usually called spontaneous revision, for making changes in persuasive essays they produced. The other group of 13 students had their essays analyzed using nine final-product analytic procedures from *Writer's Helper*. These procedures were (a) number of words per sentence, (b) number of words in a paragraph, (c) readability index, (d) word frequency count, (e) listing of sentences, (f) outlining the document, (g) usage analysis, (h) homonym analysis, and (j) "to be" usage.

Each student turned in both the first, unrevised draft and the second, revised draft. The two drafts were then compared for types of revision changes using Faigley and Witte's (1981) revision taxonomy. It was found that the essays of those students using *Writer's Helper* had significantly more revisions of the following types: (a) meaning-preserving substitutions and distributions; and (b) microstructure additions, substitutions, consolidations, and distributions (see Table 2).

Certain *Writer's Helper* revision features are highly related to the types of changes the *Writer's Helper*-group produced more often. An analysis of the feedback this group received showed that nine of the 13 had written above their intended audience. The Readability Index, for example, informed these nine writers that they were writing above their

Table 2.
Revision Changes in Essays by Students Revising Their Essays Spontaneously and Students Revising Their Essays Using *Writer's Helper*

Revision Type	Non-*Writer's Helper*	*Writer's Helper*
Meaning-Preserving Changes		
Substitution	6.769	9.231
Distributions	3.077	5.077
Microstructure Changes		
Additions	1.769	3.923
Substitutions	2.154	4.154
Consolidations	4.692	8.385
Distributions	4.154	7.0

audience—10th graders—and one way to remedy this potential comprehension problem would be to shorten sentences, or to make two sentences out of one existing sentence. This revision-direction might have been prompted by the Number of Sentences feedback they received. Twenty-seven percent of the sentences that ten of the 13 wrote were more than 35 words long; they received the message that some of their sentences were considerably longer than the rest and they might want to shorten them. Both the Readability Index and the Number of Sentences procedures may have prompted more Distribution, converting one sentence into two or more sentences. Meaning-preserving Substitutions may have been due to the Homonym, "To Be," Usage, Gender-related Words, and Word Frequency Count procedures, all of which prompted viewing words and pondering changes.

As stated earlier, writers in the *Writer's Helper*-group produced significantly more Microstructure Additions, Substitutions, Consolidations, and Distributions. Microstructure Additions are those complete textual units that did not exist in the first draft and that bring new information to the text. Substitutions are those textual units that replace ones existing in the first draft and change the meaning at that level. Consolidations are occurrences of combining sentences and changing the meaning. Distributions are occurrences of turning one sentence into two or more. Although consolidations and distributions involve working with existing information, their transformation resulted in meaning change. For example, subordinating one sentence to another—an instance of consolidation—might have changed the meaning by establishing a causal relationship between two previous, unrelated sentences. Or, the use of semicolons and colons might have strengthened the relationship between two sentences that was not previously as apparent. On the other hand, separating units within a sentence and creating more than one sentence might eliminate a relationship between textual units, thus establishing a different meaning. The *Writer's Helper* revision features that might have prompted these kinds of changes are Number of Words in Sentences, Number of Words in Paragraphs, the Readability Index, Listing of Sentences, and Outlining the Document.

There were nonsignificant differences in the number of the following types of revision: (a) **formal** spelling, punctuation, tense, abbreviation, and number changes; (b) **meaning-preserving** addition, deletion, permutation, and consolidation changes; (c) **microstructure** deletion and permutation changes; and (d) **macrostructure** changes. It makes sense that the two groups—comprised of good writers—might not differ in the number of **formal changes** because they should have the inherent skills to make such changes on their own and there was nothing in the *Writer's Helper* procedures that would have had them focus on such changes. **Macrostructure** changes, likewise, would not be prompted necessarily by the procedures, mainly because macrostructures are based on affecting major sections of the essay. Although writers of both groups did add, delete, substitute, permute, distribute, and consolidate macrostructures, the differences between the two groups were not significant.

The more confusing points center on the reasons the *Writer's Helper* procedures did not cause more **meaning-preserving** addition, deletion, permutation, and consolidation changes and **microstructure** deletion and permutation changes. One reason for not effecting **meaning-preserving** addition and consolidation change-differences may be that, when they added and consolidated, the meanings were changed; in that case, they were classified as microstructure changes. The sentence-analysis, the paragraph-analysis, and the readability index have been identified as possible prompts for these kinds of changes. The lack of difference between the deletion changes of the two groups may have been due to their being "absorbed" by the consolidation/microcostructure processes since consolidation—combining two sentences—might result in eliminating words; but, if the consolidation has been explained as microstructure changes, then the differences would not appear as **meaning-preserving** changes. There is nothing evident in *Writer's Helper* that might prompt permutation changes—rearranging words. Writers from either group produced only a few: the mean for the *Writer's Helper* group was 3.692 (sd= 1.653); for the non-*Writer's Helper* group, 3.231 (sd = 2.006).

The same line of reasoning may be applied to the lack of difference in **microstructure** deletions: most likely the differences were "absorbed" by the significant **microstructure** consolidation changes. **Microstructure** permutation changes did not significantly differ between the two groups again because there is nothing in *Writer's Helper* that would necessarily prompt that kind of change. Similarities between the two groups support the notion that both groups would rely on their existing permutation-related skills.

The quality of the essays of the two groups was compared; those students using the *Writer's Helper* revision analysis produced higher quality essays. It was also found that of the 23 revision types, only three reliably predicted quality: **microstructure deletions, microstructure distributions,** and **microstructure consolidations. Microstructure deletions** and **distributions** had positive relationships with quality—that is, the more deletions and distributions that changed the meaning of the text that a writer performed, the higher the quality. On the other hand, there was a negative relationship between **microstructure consolidations** and quality. The more consolidations that changed the meaning of the text performed by the writer, the lower the quality; the sentences may have been too long and difficult to understand.

If revision is viewed as a sign of good writing, then the revision procedures of *Writer's Helper* seem to promote such activity. Certainly some caution needs to be exercised because not all of them were significant differentiators. As stated before, such lack of differences may direct teachers to revision units they may need to instruct.

The Integration of Writer's Helper in Writing Instruction

In another study, I (Reed, 1990b) found that the use of *Writer's Helper* prewriting prompts were effective when writing different types of writing, or modes of discourse. The 18 college students first wrote a descriptive essay by hand, followed two weeks later (or week 3) by an expository essay after they had been exposed to the *AppleWorks* word processing program. Then four weeks later (or week 7), they wrote a narrative essay using the Teacher's Questions prewriting program; I had programmed in 10 prompts based on the Reporter's Formula (the who, what, where, when, why, and how of the event). Four weeks later (or week 11), the students wrote a persuasive essay using the Debate an Issue prewriting procedure. And then finally during week 15, they wrote another descriptive essay by hand.

I found that this instruction, which was comprised of the five writing tasks, was effective; the quality of the students' essays did improve from pre-treatment to post-treatment. An additional finding was that the students' writing performance did not regress once the computers were removed. Other pre-treatment/post-treatment findings were that writing apprehension did not decrease from pre-treatment to post-treatment; however, computer anxiety did.

But, other interesting findings were due to the repeated measures design which allowed me to "track" the internal performance and attitude measures while completing the second, third, and fourth writing tasks. Writing quality increased when, after writing the first two tasks—a descriptive and an expository—they wrote a narrative; but the quality then decreased when they wrote the fourth task—a persuasive essay. The quality of the next task—also a descriptive essay—increased once again (see Table 3).

The changes in quality within the treatment may have been due to one of three factors: (a) mode of discourse, (b) improved use of the computer as a writing tool, and (c) general improvements in writing. All three reasons seem logical for the narrative essay: (a) narratives are perceived as being the easiest and most-often written mode; (b) the student-writers had been using computers for six weeks at that point; and, (c) they had been writing for the six previous weeks. The decrease in quality when writing the persuasive essay probably was due more to the mode of discourse; persuasions are considered to be the most difficult because they are written the least often and have as their primary purpose convincing an audience to change his or her mind, rather than the purpose of a narration which is to recount an event. This reasoning seems to be supported by the increase in writing apprehension when writing the persuasion. Their heightened apprehension was a function of writing task and its relative degree of difficulty. Because their computer anxiety remained low and stable, the lower performance was not due to their anxiety toward the computer. The further increase in quality when they wrote the descriptive post-treatment essay must be related to mode of discourse; because they wrote this final task by hand, it was not due to any fears toward using the computer which continued to remain low and stable. Their writing apprehension also decreased from the persuasive task to descriptive task.

In summary, *Writer's Helper* prompts used for the narrative and persuasive essays were easily integrated in computer-based writing instruction. The findings were in line with other research on the use of modes of discourse in non-computer-based writing instruction (i.e., Flower & Hayes, 1984; Reed, Burton, & Kelly, 1985). Although the student's writing apprehension did not decrease from pre-treatment to post-treatment, it did fluctuate when writing persuasive essays. Perhaps more importantly, the quality of their writing did increase as a result of the treatment, and their fear toward the computer did decrease early in the instruction and remained stable and low. It is pertinent that the computer, when used in instruction, is not

Table 3
Means and Standard Deviations of Quality Scores, Writing Apprehension, and Computer Anxiety

	Week 1 Desciptive	Week 3 Expository	Week 7 Narrative	Week 11 Persuasion	Week 15 Descriptive
Quality	5.11	5.17	6.56	4.83	6.06
	(.76)	(.99)	(.86)	(.99)	(1.00)
Writing Apprehension	46.11	50.61	50.11	57.5	47.89
	(10.35)	(10.34)	(11.03)	(11.46)	(11.40)
Computer Anxiety	45.56	49.89	37.67	34.17	33.61
	(11.09)	(12.24)	(8.51)	(8.69)	(9.66)

perceived as something to be feared; otherwise, it will interfere with the cognitive objectives of the instruction—in this case, improving writing.

Summary

The use of composing process software seems to promote different kinds of composing activities not provided by word processing alone. The prewriting features, specifically those of *Writer's Helper*, seem to be most useful for low-ability writer. Teachers of below-average writers should consider integrating such procedures to help their students overcome such deficiencies. Average writers probably will likewise benefit. The point of departure centers on above-average writers who, for the most part, will not benefit as much. This makes sense because the reason they are good writers is because they already have these skills. The teachers of these students should probably not impose these prewriting prompts.

But regardless of writing ability, the revision-related procedures of *Writer's Helper* seem to benefit students of all abilities. Several of the analyses predict final-product quality and syntactic complexity, as well as prompting different types of revisions. Writing instructors have long claimed that good writers revise more than poor writers. If such software promotes and guides revision, then its use appears to be pedagogically sound. Certainly the findings of the two studies explained in this paper attest to that.

Given the findings of the three previous studies, it is promising that *Writer's Helper* can be easily integrated in commonly occurring writing instruction. Although writing apprehension may not necessarily decrease, computer anxiety apparently does after a brief period of time. Fluctuation of essay quality and writing apprehension may be due more to difficulty of writing task than the use of composing process software-guided writing tasks. However, the generally well-known benefits of computer use can enhance such instruction.

In general the four studies described in this paper support the notion that computer use in writing instruction can be extended beyond just word processing. Composing process software adds needed dimensions; in many cases, these additional dimensions produce better writing than word processing alone.

References

Emig, J. (1971). *The composing processes of twelfth graders.* Urbana, IL: National Council of Teachers of English.

Faigley, L. & Witte, S. (1981). Revision analysis. *College Composition and Communication, 32.*

Flower, L. & Hayes, J. (1981). A cognitive process theory of writing. *College Composition and Communication, 32.*

Flower, L. & Hayes, J. (1984). Images, plans, and prose: The representation of meaning in writing. *Written Communication, 1,* 120-160.

Hannafin, M. & Peck, K. L. (1988). *The design, development, and evaluation of instructional software.* New York: Macmillan Publishing Co.

Hawisher, G. (1987). Research update: Writing and word processing. *Computers and Composition, 5,* 7-27.

Perl, S. (1979). The composing processes of unskilled writers. *Research in the Teaching of English, 13,* 317-337.

Reed, W. (1988). *Suggested appropriateness for the prewriting features of* Writer's Helper. Presented at the annual meeting of the Eastern Educational Research Association, Miami.

Reed, W. (1989). The effectiveness of composing process software: An analysis of *Writer's Helper*. *Computers in the Schools, 6*(1).

Reed, W. (1990a). *The effect of composing process software on the revision and quality of persuasive essays.* Presented at the annual meeting of the Eastern Educational Research Association, Clearwater, FL.

Reed, W. (1990b). The effects of computer-based writing instruction and mode of discourse on writing performance and writer anxieties. *Computers in Human Behavior, 6,* 211-221.

Reed, W., Burton, J., & Kelly, P. (1985). The effect of writing ability and mode of discourse on cognitive capacity engagement. *Research in the Teaching of English, 19,* 283-297.

Reed, W. & Sherman, T. (1990). Using memory during prewriting: The comparative effects of external storage and idea organization on the syntactic complexity and quality of basic and honors writers' narrative essays. *Journal of Research and Teaching in Developmental Education.*

Sommers, N. (1980). Revision strategies of stgudent writers and experienced writers. *College Composition and Communication, 31,* 378-388.

Stallard, C. (1974). An analysis of the writing behavior of student writers. *Research in the Teaching of English, 8,* 206-218.

Wresch, W. (1984a). *The computer in composition instruction: A writer's tool.* Urbana, IL: National Council of Teachers of English.

Wresch, W. (1984b). Questions, answers, and automated writing. In W. Wresch (Ed.), *The computer in composition instruction: A writer's tool.* Urbana, IL: National Council of Teachers of English, 143-153.

Wresch, W. 1987). *A survey of computer preparation in English education programs.* Paper presented at the annual fall meeting of the National Council of Teachers of English, Los Angeles.

Projects
Music and the Arts

Logo, Art, and Architecture in the Middle School

Chuck Friesen
Lincoln Public Schools, Nebraska

Abstract

This session will describe a middle school project that integrated the study of art and architecture with Logo programming. The works of art served as the motivating force for the scope and sequence of a Logo programming unit. Throughout the project emphasis was placed upon top-down design and structured programming techniques.

Polya's four step problem solving model was used as a model for the students as they worked on their programming projects. In addition, these problem solving skills were emphasized and integrated into the students' work: make a simpler problem, guess and check, pattern identification, working backwards, trial and error, and the use of similarities and differences.

A description of the Logo commands, concepts, and procedures the students used to complete their projects will be shared. In addition, examples of student projects will be presented.

Enchanted Amiga Room: Programmed Take-Off on Disney's Enchanted Tiki Room

Marla Weiss
Community School of Naples, Florida

Abstract

At Disney World, one exhibit is the Enchanted Tiki Room in which mechanical birds entertain the audience by talking and singing. At the Community School of Naples, sixteen middle school programming students created a multi-media event called the "Enchanted Amiga Room" — a programmed take-off on this attraction with talking computers. With clever use of variables and looping, and the Amiga's 4096 colors and speech and sound capabilities all accessible through BASIC, the students created twenty minutes of magnificent entertainment, totally programmed.

Combining Graphic Arts and Animation

Jim Madeheim
Sahuarita, Arizona

Abstract

Abstract unavailable at press time.

Projects
Multimedia Teacher Education

Extending Multimedia Computing Through Distance Learning

Nancy Dill
Queens College, City University of New York

Abstract

This presentation will review knowledge navigation strategies for the 21st century. Specifically, it will lool at how converging computer and telecommunications technologies are shifting the control of multimedia computing to individuals who access and configure the dimensions of their own learning.

This presentation will consider federal information infrastructure legislation (drafted by Senator Gore and others) and argue for an integrated computer/television (broadcast/cable/satellite) system, open to a variety of programming/networking vendors.

Distance learning issues, such as copyright clearances for video segments and difficulties with "certification" demands across statelines, will be discussed.

Finally, the presentation will consider the role Distant Learning systems can play in the development of video databases (e.g. teaching performance/pupil progress portfolios) and consider the qualitative research dimensions opened up by Distance Learning Technologies.

Linking Member Groups with ASCD's Electronic Network

Vicki Hancock
Association for Supervision and Curriculum Development
Alexandria, Virginia

Abstract

The Association for Supervision and Curriculum Development has added a high-tech component to its member services - a computer conferencing system. The presenter will outline hardware and software components, discuss the system's capabilities, describe how member groups have used it so far, and delineate plans for future applications.

The system provides three major capabilities. The electronic mail component allows participants to exchange private messages with each other. The bulletin board component facilitates distribution of information from the host computer or from any participant to all the members of a conferencing group. The conference component permits participants to create ongoing recorded discussions which are indexed by title, author, and subject categories. File transfer and full text searching enable users to exchange and find text containing any word or phrase they specify.

Anticipated applications of the system include expanding the system to include a wider variety of ASCD members and member groups, using the system as a real-time resource for updatig members on education-related topics, and initiating manuscript and proposal submissions for publications and conferences.

A National System for Electronic Transfer of Student Records

Robert Friedman
Florida Department of Education

Abstract

A national system of electronic transfer of educational records for students in gradess prekindergarten through 12 and in postsecondary education.

An effort is well underway to design, test and implement a nationwide system for electronic transfer of educational records. The NCES has funded a task force of school districts and state education agency personnel from California, New York, Florida, Texas, and Washington state to define the requirements and procedures for electronic transfer of grades PreK-12 student records between districts and postsecondary institutions. The task force has developed a prototype system which is being piloted through July 1, 1991. The American Association of Collegiate Registrars and Admission Officers is also developing a system for electronic transfer of postsecondary transcripts. The two task forces are jointly preparing a proposed Electronic Data Interchange (EDI) standard which has been submitted for approval to the American National Standard Institute (ANSI). The presentation will describe the proposed information requirements and the ANSI EDI standard.

Projects
Science Across the Levels

Using Computers for Field Work in the Southwest/West Desert Environment

Jim Anton
Phoenix Country School, Arizona

Abstract

The project presentation instructs teachers how to use computers for the implementation of field work. This project is made possible because of a *HyperCard* program Mr. Anton developed entitled *MacDesert.*

Students involved in the project conduct field work, identify Southwest flora, conduct experiments or activities in the field and in the laboratory, propagate and cultivate native flora, and finally learn how to conduct research and write a science research paper on Southwest/West desert ecology.

Teachers are also instructed how to easily expand and/or modify *MacDesert* in ways to make it specifically suitable to their own teaching situation whatever it may be. Teachers participating in the project presentation will be shown how *MacDesert* may be used to identify and find out more about unknown cacti.

The project presentation is for teachers from middle school through high school. However, teachers in other disciplines may also find the program useful. For instance, teachers interested in interdisciplinary studies can find out facts about Native American myth and/or studies, as they relate to other facts uncovered in field work.

Interactive Multimedia Social Science Simulation Project "Flatnoid"

G. Beytin, M . Elowitz, L. Elowitz
Hamilton Humanities Magnet
Los Angeles, California

Abstract

This project has five goals: (1) To create a complete and self-contained environment in which students can learn social science concepts and skills in several learning modalities simultaneously. A few of the many skills supported are: political argumentation, oratory, the practical application of political and economic theory, democratic procedures, and journalism. (2) To provide a context in which the development of the related skills of expository writing, debate, oratory, project planning, and graphic presentation are integral to participation in the simulation. (3) To provide an environment in which the more problematical student can achieve and interact in a positive way with his/her peers. This type of involvement will result in higher self-esteem for many students that are lacking sufficient successful intellectual experiences. (4) To teach computer skills indirectly as a facet of participation in the simulation. This way, students will come to see computers and related technologies as the gateway to more effective and/or competitive participation in Flatnoid. This, in effect, simulates the role new technologies, such as the microcomputer, play in the real world—as a tool helpful in the more efficient completion of complex tasks. (5) To provide a context for the discussion of related topics in the Humanities. For example, a student debate on citizen voting rights (i.e. one team—one vote, one population unit—one vote, or one student—one vote) that is often the first order of business for students writing a constitution for one of the countries of Flatnoid, can be directly related to a unit on the Constitutional Convention in a Government class.

Flatnoid is a simulation of life on a fictional planet. It is played by student teams assigned the role of a social class in a country made up of other teams, or in some cases, an entire country. Teams make all the critical decisions of where to invest, whether investment should be intensive or extensive, social or military, in agriculture or industry, and so on. After all teams have made their moves, the various moves (through sneaker net or LAN) are collected and processed by a central program which then alters the team files to reflect the interactions that have taken place. Some interactions may be cooperative, some may be written or graphic communication, and some may be coercive.

Over the past five years Flatnoid, in various versions, has been successfully played by the students of Alexander Hamilton Humanities Magnet High School (a part of the Los Angeles Unified School District). Until this year, Flatnoid was a paper-based game inherently limited by its complexity and logistics. The computer provides a central nexus upon which simplicity and greater depth can be applied in the design and execution of the simulation. Thus, with the access to interactive software the students now enjoy, the real potential of this interdisciplinary and interactive curriculum concept can be realized.

One of the meeting places for the participants of Flatnoid is the Amiga computer. The Flatnoid Software is based on AmigaVision, a multimedia package published by Commodore Business Machines. The Amiga, and AmigaVision software, provide a cost-effective, powerful, and decentralized solution for a multimedia project such as this. In addition, there is available for the Amiga, several good word processing, paint, accounting, and DTP packages that easily integrate with AmigaVision. The Amiga's multitasking operating system and inter-task communication provide an opportunity to integrate other software directly into the workings of the simulation. This allows for a high degree of flexibility and multimedia creativity for the scenario developer, classroom teacher, and the simulation participant. This is not just hypermedia software controlling another device such as a laser disk, but a extension of the usual meaning of multimedia to include the entire range of tools the microcomputer provides us with.

The presentation will include a demonstration of the Flatnoid software as well as examples of the above described extended multimedia environment. Participants will be provided with extensive documentation and a sample of the Flatnoid software.

Projects
Bilingual Computing

Creative Computing with ESL/LEP Students

Sue Espinoza and Judi Repman
Texas Tech University

Abstract

One area of growing interest today is the use of computers with K-12 students learning to speak English as a second language. These students may lack native language reading and writing skills that would transfer to their new language, and they many also lack motivation or interest in acquiring a working knowledge of the English language.

This presentation will discuss a variety of teaching strategies that have led to improved motivation and learning by these students. Examples of ESL-specific and non-specific cross-curricular applications will be demonstrated. Software programs (such as *Crossword Magic*, *Grammar Gremlins*, and *Word Wizard*) that have been adapted for ESL/LEP classroom use will also be presented. Videotaped examples of ESL/LEP students' products created with these programs will be shown and participants will be provided with a handout listing suggested software and sample lesson plans for the incorporation of this software into ESL and regular classroom instruction. A bibliography will also be provided.

Building Connections: Students and Hypermedia

Ann Dana and Marianne Handler
Computer Consultants
Hinsdale, Illinois

Abstract

Hypermedia is often presented as a tool for teachers to use in the presentation of curricular content. It is important not to loose sight of the potential of hypermedia as a tool for students. Teachers and students need to understand the importance of incorporating the design elements of visual literacy into the planning and implementation of hypermedia projects. Teachers provide students with the tools they need to clearly communicate their ideas when they speak and when they write. Knowledge of these design elements will supply them with new tools for clear communication when using hypermedia.

In support of this assumption our presentation provides examples from projects created by students in grades three through eight during this school year. These examples focus on curricular content and students' understanding of the design elements applicable to their projects.

If It's Saturday—It Must Be Computer Check-Out Day!

Diana M. Willis
Suwannee District Schools, Florida

Abstract

As part of an ongoing effort to have our Chapter I parents involved in their child(ren)'s education, Suwannee County began a Computer Home Loan Program in February of 1990. This program brings parent(s) and child(ren) to school on Saturday mornings where together they learn how to set up and operate an Apple IIc Plus. Software is available for preview prior to selection for home use. The emphasis is on parent-child "togetherness," improved communication between parent and child, and active participation by the parent(s) in the child(ren)'s learning process. In addition to being well received and highly rated as a successful and productive program by both the parents and the children who have participated, this program allows teachers to meet and talk with parents in a non-threatening atmosphere.

The presentation will share how the program evolved and what happens on a typical Saturday morning. Information will be provided concerning the status of the project and the future plans for this worthwhile venture.

Projects
Teaching Global Education via Telecommunications

Learning Circle Projects Sponsored by Global Partners

Margaret Riel
Adel Labib
AT&T Learning Network

Abstract

Computer telecommunications (telecomputing) helps students reach beyond their classrooms to people, places, and events anywhere in the world. The barriers between classrooms are crumbling like the Berlin Wall. Teachers and students in different countries use their classroom computer to work cooperatively exploring issues from both local and global perspectives.

In Learning Circles on the AT&T Learning Network, teachers and students design curriculum-based projects. Each participating class designs or "sponsors" one of the Circle projects. In this presentation, we describe the projects that were sponsored by students in countries other than the United States. Currently there are 23 countries involved in the AT&T Learning Network. We will describe the projects sponsored by students living in Saudi Arabia, the Soviet Union, Italy, and many other locations.

This past year, for example, students in the east and west of Germany designed a project to explore the economic problems facing their newly unified country and the effect of these problems on the global economy. The students in France explored strategies for improving schools. Participating in these projects gives students a richer understanding of local and global issues.

Telecommunications in a Global Community

Virginia Lawson
Mimosa Park Elementary School
Luling, Louisiana

Abstract

As we embark into the twenty-first century, educators must realize the importance of incorporating computer technology into the core curriculum. Educators in St. Charles Parish, Louisiana, have done this through a global telecommunications project entitled Kid Link: Learning On Line. Telecommunications is integrated into the areas of Language Arts, Science, and Social Studies.

This presentation will inform participants how to effectively and successfully establish telepartnerships with classrooms around the nation as well as the globe. Steps for setting up national and international partnerships will be outlined with emphasis on the following goals:

- Incorporate technology into the curriculum
- Access information on current events and topics for research
- Exchange, conduct, and compare results of science experiments
- Learn of the diversity among peoples by sharing cultural information
- Becoming active partners in the world community by thinking globally while acting locally

Students in elementary classrooms were exposed to new avenues of learning through the process of telecommunications. Working with cooperative group settings within their respective classrooms, they conducted research, wrote reports, collected scientific data, and shared diverse social and cultural information. Technology was used by teachers to enhance activities that enabled students to become active, collaborative learners while using hands-on activities-investigating, inquiring, and discovering new knowledge. Students became self-confident, independent learners and developed critical thinking skills necessary to become skilled, productive workers in the twenty-first century.

Kid Link is a project that has been formed among elementary schools within the United States, and most recently expanded on a worldwide level, using one elementary classroom as a pilot to test the practicality of transatlantic communications. Specifically, a Russian partnership was formed for this means and the steps involved in establishing this partnership will be reviewed. Successes and cautions will be shared..

We Are One World: A Multicultural Telecommunications Project

Ann Grady
Boston Public Schools
Boston, Massachusetts

Abstract

"We Are One World" is a multicultural telecommunications project involving 600 Boston Public School students in grades K-5. Twenty five classes in 20 schools collected data on where their parents were born, organized their survey results using charts and graphs, and used telecommunications to share their results with the other classes participating in the survey. Their data revealed that students' parents were born in 40 countries and 26 states. The project generated spin-off projects in math, social studies, writing, literature, art, and cooking.

Goals

The project was developed to introduce students and teachers to telecommunications as a vehicle for sharing information. Project organizers also chose the project for its multi-cultural focus as well as its rich source of data analysis opportunities relevant to students' personal experience.

Methodology

Each of the twenty five teachers participating in the project was provided with an Apple IIe computer with a modem and phone line as part of Boston Public Schools' contribution to a National Science Foundation Teacher Enhancement Grant.

The teachers met to define the topic and to work out details of its implementation. As a result of this discussion, the initial questions, "Where were you born?" was revised because teachers felt that the survey results for this question would be less interesting (ex. all of the children in a Chinese bilingual kindergarten class were born in Boston!). After long discussion, it was also decided to ask the question, "Where was one of your parents born?" to reduce student apprehension about how to answer the question if they lived with only one parent.

Survey forms and procedures were established and the parent letter translated into Spanish, French, and Chinese. Student and parent interest in the survey was very high. Teachers reported hearing from parents who had not responded to any previous request for information. One parent responded that this was the first time anyone had asked her about her culture! Parents shared recipes, stories about holiday traditions, favorite books, etc. Many teachers invited parents to visit their classes to share their experiences in person.

Students created bar graphs using "Post-It" notes and construction paper. Kindergarten and first grade students easily read their class bar graphs and proudly shared their results with visitors.

Telecommunicating the survey results proved more difficult. At the time of the survey, the project was using a small Bulletin Board operating with one phone line and an Apple II computer. Some teachers used Appleworks to create a database for their information while other teachers sent their data on E-Mail. Each teacher sent his/her information to a "Captain" who collected the data for five classrooms. The Captain then organized the data for the five classrooms and sent the compiled data to the "Super Captain." The Super Captain's fourth grade class organized the data for all classes, compiled charts and graphs and sent the result back to each class. The charts and graphs compiled by the fourth graders were shared with all teachers at an evaluation session.

Plans For Future Implementation

A similar survey will be conducted during the 1990-91 school year. Boston Public Schools now has its own multi-line bulletin board with up-loading and down-loading capabilities. Training in the use of Appleworks database will be provided to each participating teacher. All data will be organized using Appleworks and uploaded to a Captain. The compiled data will then be available to each teacher and classroom.

Special Presentation

WAR IN AI: What Does it Mean for Educational Computing?

Seymour Papert
Massachusetts Institute of Technology

Connectionism, Neural Net, Parallel Distributed Process. The Society of Mind. Emergence, Actors, Agents, Artificial Life... There is a growing list of words no one heard in computer science courses five years ago, but which today challenge the key paradigms of the discipline.

Time will tell how far the challenge will go. But it has already gone too far to be ignored in Educational Computing.

This presentation will be half tutorial, half polemic, and half nuts and bolts advice for educators.

Special Presentation—ECMI

Funding Computing Research and Instruction at Minority Institutions

Jesse C.Lewis
Norfolk State University
Norfolk, VA

Panel discussion covering ways and means of funding computing research and instructon at Minority institutions. The discussion will include reports on on-going projects and available resources for instruction.

Special Presentation—ACH

What are Networked Information Resources and Where to Find Them

George Brett
The UNC Educational Computing Service
Research Triangle Park, North Carolina

This session will explain about networked information resources. It will show examples taken from telecommunications sessions. Strategies for locating and accessing these resources will also be presented. networks to be discussed will include: The Internet, BITNET, USENET, NSFNet, and others.

Thursday

NECC '91 CONFERENCE PROCEEDINGS

Special Presentation
The Editing of Reality: Desktop Video & Multimedia Composition as Basic Skills for the 90's

Ted Kahn
Digital F/X, Inc.
Mountain View, California

The worlds of computers and television are colliding. The integration of graphics, typography, video, audio, and animation will not only change the way in which each of these media is presented, but also how they are conceived and produced. While the use of video is ubiquitous in corporations, education, and the home, most of us are still just consumers of video and television programming. But this situation is rapidly changing. Desktop video and multimedia composition tools are now giving students and teachers the capability to become producers, as well as consumers, of professional quality programs.

In order to make best use of these technologies, students and teachers will need a new set of basic communications skills. Perhaps the most important conceptual skill is that reality in these media is *edited*, i.e., molded together, from many separate pieces.

This special session brings together outstanding educational researchers & teachers who have already pioneered the integration of computer-based animation, video production and multimedia composition into the K-12 curriculum. The purpose of the symposium is to discuss their experiences and thoughts regarding supplementing & restructuring the curriculum to make optimal use of these technologies. The symposium will include demonstrations/sample animated films, videos, and multimedia compositions produced by students and teachers.

This session will also be followed by a "hands on" laboratory to give participants first hand video editing experience using Digital F/X's Macintosh-based Video F/X™ and Soft F/X™ desktop video production systems.

Ted M. Kahn (Manager, User Education, Digital F/X, Inc., Mountain View, CA).
New languages and grammars for video & multimedia.
Ted Kahn has been involved in pioneering projects in the use of computers and educational technology for over 20 years. As Manager of User Education for Digital F/X, he is now exploring innovative approaches for teaching video and multimedia production to students, teachers, and corporate communications professionals. Digital F/X was also recently awarded an Emmy for its outstanding technical contributions to video workstation technology.

Dave Master (Rowland High School, Rowland Heights, CA)
Animation: An interdisciplinary curriculum approach.
Dave Master is an award-winning high school teacher. Over the past decade, students in his animation lab and video production classes have won scores of international film awards. Many of his former students are now working as professional animators, designers and filmmakers for major Hollywood studios and Fortune 500 companies. His classroom also serves as Southern California's only certified adult Regional Occupational Program (ROP) in film, animation, and video production.

Enrique Godreau (Xerox PARC and the Institute for Research on Learning (IRL), Palo Alto, CA):
MediaWorks: Multimedia composition in the classroom.
Enrique Godreau conducts research on multimedia technologies, systems, and applications in the areas of document processing, communication and education. The MediaWorks project at the Institue for Research on Learning is designed to provide multimedia design, databse, and composition tools to middle school students and teachers. Godreau co-developed the system and collaborated with a middle school in the San Francisco Bay area to replicate the project at their site.

Herb Kohl (Teacher & writer, Director of the Coastal Ridge Education & Research Center, Point Arena, CA)
Video literacy.
Winner of the National Book Award and one of the longtime leaders of the open education movement, Kohl has had extensive first-hand experience in teaching writing, science, mathematics, and game design at all grade levels. He has been especially interested in the use of media—computers, film, and video—in the schools.

Special Presentation

Logo Based Hypermedia Systems for Small Computers

Michael Tempel
Logo Computer Systems
New York, New York

Abstract unavailable at press time.

Projects
Computer Uses: ESL, CAI and Family Literacy

Using Computers to Teach American Idioms to ESL Students

Mei-Yan Lu
San Jose State University

Abstract

American idioms play an important part in verbal communication. However, very few of the ESL text books emphasize the importance of idioms in the American languages. This lack of teaching idioms results in frustration of ESL students who are trying to strike a fruitful conversation with their American counterparts.

The Instructional Technology Program at San Jose State University produced a MS-DOS based hypermedia program using ToolBook to teach American idioms to its international graduate students and local high school ESL students.

The source of the content is based upon a survey of unfamiliar idioms to the first and second year international students at San Jose State University. This American idiom program is in color and runs under Microsoft Windows in the MS-DOS platforms. The program will be demonstrated in the presentation. Research results including feedback from the students and leaner control issues versus system control issues in different age groups and ethnic background will be reported and discussed in the presentation.

The Use of Word-Processing Macros to Create CAI Routines

Joel Nydahl
Babson College
Wellesley, Massachusetts

Abstract

Composition teachers and curriculum planners searching for pedagogically sound ways to use computer applications in the classroom ought to know that the word-processing program their students use can be a powerful instructional tool.

By utilizing the macro capability of many off-the-shelf word processors, composition teachers can custom-design the kind of revising and editing routines usually associated only with special CAI software.

This presentation will focus on conceptualizing, creating, and using three kinds of writing-aid macros: Active-boilerplate macros that replicate heuristics; text-manipulation macros that can alter the form of student text; and player-piano macros that record and reproduce all keystrokes executed by writers during composing sessionThe Use of TETRIS in the Mathematics Curriculum.

Family Literacy and Computers

Rebecca King
Louisville, Kentucky

Abstract

Abstract unavailable at press time.

Projects
Problem Solving

Kids Today: Adults Tomorrow—Computers in the 21st Century

Mary S. Haverfield, Brevard Williams
Gwinnett County Public Schools
Snellville, Georgia

Abstract

Today's students will face the 21st century with a variety of challenges. These challenges include an expanding technology and every-increasing competition from abroad. Job description will constantly change as technology becomes more complex. We must equip students with the tools to understand these changes. Therefore, the focus is not to learn about technology but to learn how technology is expanding in the workplace.

In this session, participants will explore methods for helping structure the roll of application programs in the job world. From the first day, students apply problem solving skills and creative thinking to develop a simulated business. This project guides students through an exploration of word processing, data base and spread sheet. Class exercises include many opportunities for cooperative learning and developing inner disciplinary linkages. Information on how to set up a similar project will be available.

Computer Database Classroom Activites: Updates From a National Study

Vivian Johnson
Montana State University

Abstract

Abstract unavailable at press time.

The Use of TETRIS in the Mathematics Curriculum

Ray Braswell
Auburn University at Montgomery

Leah McCoy
Wake Forest University

Abstract

This project used the computer game TETRIS as a means of improving students' spatial skills. Several studies, including Flake (1990), Battista, Wheatley, & Talsma (1989), and Fennema and Sherman (1977), have noted the correlation between spatial ability and mathematics achievement. It is a recognized fact that many students, particularly females, are deficient in spatial skills.

The computer game TETRIS involves manipulating geometric shapes to fill an area. The shapes are all composed of four squares in different configurations. The player is able to slide or turn the shape as it "falls" into place. This is a powerful tool for developing spatial skills as well as other geometry concepts. It is a dynamic activity; the student visualizes where the "piece" will fit, places it there, and then sees the result.

Informal observations and self-report data indicate that practice with TETRIS is related to improvement in spatial skills and also in problem-solving skills. Further study is underway to identify the TETRIS playing styles of experts and novices, and to determine long and short term effects of TETRIS experience on mathematics skills, particularly in geometry. If the preliminary findings hold true, we may find ourselves encouraging students to play certain video/computer games.

Special Presentation—SCS
Inquiry Learning in Mathematics

The panel will address a small number of general questions concerning inquiry learning in mathematics: What is the relationship between inquiry-based curriculum and mathematical content? How can student learning be assessed, if each student learns something different? How is the potential for inquiry learning limited or augmented by the availability of technological tools?

The panelists will base their remarks on their current work in the are of inquiry learning in mathematics. Tim Barclay has developed and tested several inquiry-based mathematics activities with TERC's Star Schools Project, in which students in different schools share their discoveries over a communication network. Rebecca Corwin directs TERC's Talking Math project, which is concerned with the role of discussion in mathematics classes, and she has also developed and published inquiry mathematics curriculum with TERC's Used Numbers project. Chris Hancock directs TERC's Hands On Data Project, which is concerned with the development of computer tools to support data-based inquiry. Andee Rubin was Principal Investigator in BBN Lab's Project on Reasoning Under Uncertainty, where she developed tools and curriculum for statistical inquiry, and piloted innovative use of video technology to support inquiry learning.

Panelists:

Tim Barclay, Rebecca Corwin, Chris Hancock, Andee Rubin
TERC
Cambridge, Massachusetts

Projects
Teacher Training

Administrative Staff Development

Jeffrey Branzburg
New York City Public Schools

Abstract

Abstract unavailable at press time.

Teachers and Technology: Making the Connection

Carole Martinez,
Elizabeth Bankhead
Englewood, Colorado

Abstract

This presentation will describe a Macintosh network developed for teacher productivity at Cherry Creek High School in Englewood, Colorado. The impetus for the development of this network came from opinions expressed by faculty when asked about their potential use of a proposed Macintosh lab for students. It became quite apparent that without adequate training of teachers, the lab would sit empty most of the time. The computer council recognized that any plan for instructional use of computers needed to be accompanied by an equally strong component which would support teacher use of technology.

A variety of issues related to the establishment of our network will be discussed. These include staff development for teachers, the organization of a network learning group to decentralize the technical support, network hardware and software, and network administration.

An Integrated Curriculum Model of Computer Education for Elementary Teachers

Kathleen Skinner, Christine Nelson
City of Manchester, New Hampshire School Department

Abstract

The project presentation will focus on the adaptation of the National Geographic Kidsnet Project as a vehicle for in-service education of elementary teachers. The design of the project was based on the need for elementary teachers to experience the use of computers in science and mathematics as a collaborative effort in search of answers to real problems. Teachers were trained in methodology, content and telecommunications.

The presentation will include an overview of the project; design of the inservice training module; specifics on the logistics of the telecommunication component and teacher outcomes in terms of confidence in teaching science, use of the computer as a tool, reaction to the materials and "spin off" effects.

The project was funded by an Eisenhower grant from the N.H. State Department of Education. The training model was designed for teacher education in the Manchester, New Hampshire School District utilizing the Kidsnet components produced by theNational Geographic Society.

Projects
Current K-12 Educational Technology Research Activities

Several research and development laboratory managers from the Texas Center for Educational Technology will discuss results of year-long research activities exploring innovative and experimental uses of technology in K-12 education.

Partial list of panelists:

George Culp
Site Coordinator, University of Texas at Austin
Manager, Collaborative Software Development Laboratory

J.L. Poirot
Co-Director of TCET
Manager, Curriculum and Development Laboratory
University of North Texas

Cathleen Norris
Manager, Teacher Productivity and Training Laboratory
University of North Texas

Gerald Knezek
Manager, Educational Telecommunications and Informatics Laboratory
University of North Texas

Projects
Computers are Good Listeners: Deaf Education

Enrichment Computer Camp for Communications Exceptionalities

John M. Smith
Centre for Curriculum Resources and Technologies, Ontario

Abstract

The first Enrichment Computer Camp for Communication Exceptionalities (ECCCE) was held at the Robarts School for Communication Exceptionalities in London, Ontario early this summer. It provided an opportunity for fifteen deaf and fifteen learning disabled students, ages twelve to sixteen, to be immersed in a high-tech environment designed for cooperative learning. The students were exposed to multiple platforms including IBM PS/2, Apple Macintosh and the UNISYS ICON to investigate various aspects of multimedia.

The lab activities were both varied and sophisticated. Students were able to compose and produce music using Musicland on the ICON. The software is unique as it provides the students with an opportunity to experience music "visually"; drawings on the screen having both a treble and base clef are translated into musical sounds. Each group of eight students collaborated inproducing an original piece of music.

An opportunity to integrate a new Ontario Educational Software Service release, Puddles to Pondwater (September 1990), with telecommunications and an actual pond study to provide an unique experience for students as well as test the effectiveness of the software with children. The activity incorporated four sites, three of which were linked electronically (telecommunications and LAN), to investigate and identify the creatures which inhabited the pond.

The camp was generously funded by the local Chamber of Commerce and Provincial Chapter of IODE; hardware and personnel was provided by IBM, UNISYS and the Ministry of Education. These groups readily appreciated our vision to allow students with communication exceptionalities to experience enrichment opportunities, rarely afforded them, using state-of-the-art equipment and software. Through these experiences, we know that our students will acquire many of the skills necessary to put them on equal footing with their peers in the job market.

Computer Networks as Tools for Supporting Writing Development in Deaf Students

Babette Moeller, Bank Street College of Education
Carol Reich and Laura Bell, Lexington Center

Abstract

The Literacy Network project is a unique model program that integrates computer network technology into subject matter learning to enhance the development of literacy skills in deaf students. A networked system of computers was equipped with an electronic mail system and introduced into the middle school earth science curriculum at the Lexington School for the Deaf. Four groups of students, at different grade levels (7/8th, 8th/9th) have been using the system in their science work to communicate with each other and with their teachers in writing. With the support of the network technology, new learning environments have been created that allow deaf students to practice writing in a meaningful and communicative context.

In this presentation we will discuss the results of an evaluation that examined the impact of the Literacy Network model program on the development of deaf students' writing skills. Our preliminary results are very promising and suggest that there are important improvements in writing performance among students who participate in the model program for the period of a school year. We will discuss how different features of the networked learning environment (e.g. the extent to which the network is used communicatively) contribute to students' writing development. We will conclude with a discussion of network-based activities that are most successful in promoting written literacy skills in science.

"Talk to Me," Kids, Computers, Video for Whole Language Telecomputing

Debra Freedman
St. Josephs College

Abstract

Abstract unavailable at press time.

Paper

Using Computers and Telecommunications in the Elementary School

Gloria Cathcart
Malmo Elementary School
Edmonton, Alberta, Canada

Abstract

What happens when children have a computer at their fingertips for an extended period of time? What effect does the ability to touch type have on young children? How do children take advantage of the opportunity to telecommunicate on a regular basis with other children around the world?

This paper attempts to answer these and other questions. Answers are provided by the students as well as the teachers. All grade 3 and 4 children at Malmo Apple Centre for Innovation, in Edmonton, Alberta, have a computer on their specially designed desks for at least half of every day. These students are taught touch typing at the beginning of their grade 3 year, and by the end of that year most are typing at a rate about 2 and 1/2 times as fast as their pencil and paper rate. The computer is the writing tool for almost all language arts work. The computer also has a predominate place in the mathematics, social studies, science, and art programs.

The ability to telecommunicate with about 150 schools around the globe has been very thrilling and extremely beneficial to our educational program. Our students gain a greater awareness of others as well as becoming experienced, efficient computer users. Inuit students on Baffin Island have provided their own answers to the questions posed by Edmonton students about the life of the Inuit today. Children in our Arabic bilingual program can communicate with schools in the Middle East. Information on the lifestyles of children in Japan, Australia, New Zealand, Chile, and the European counties, as well as Canada and the United States is only a link away.

Background

Our setting

Malmo Elementary School, located on the south side of Edmonton, is a small urban school serving a middle class residential area and the University of Alberta married students residence. This results in students from a variety of multicultural backgrounds. We currently have about 330 students in kindergarten to grade 6.

Malmo Apple Centre for Innovation

Sponsored by Malmo School of the Edmonton Public Schools and Apple Canada Education Foundation (ACEF), the Malmo Apple Centre for Innovation (ACI) has been operational since September 1987. Directed by Mrs. Gloria Cathcart, the project, "Elementary School Integration of the Microcomputer into the Curriculum," was awarded a configuration of Apple IIgs microcomputers and some Apple software by ACEF. The contribution of Malmo School included the physical site, appropriate classroom furniture, electrical wiring, additional software, and teacher released time for the project. Mrs. Cathcart serves as project director, grade 3/4 teacher in the project classroom, and the author of materials used in the project. She also provides inservice and information assistance to the staff and the many visitors that frequent the Malmo Apple Centre for Innovation. After two years of successful operation of the initial classroom which served a morning class and an afternoon class, a second classroom was set up.

The success of this ACI model was facilitated by cooperation between the University of Alberta and Malmo School. Instruction, assistance, and research in the keyboarding program was provided by Dr. Sandra Ubelacker, professor in Secondary Education (business). Dr. George Cathcart, professor in Elementary Education, assisted in the testing and research. Considerable technical support for the operation of the file server was provided by Mr. Ray Lauti, microcomputer consultant with Edmonton Public Schools.

Purpose

> I like the computers because it's a lot easier than using a pencil and paper. And if you make a mistake on paper it might be a lot harder to correct than on a computer. The grade three room has lots of neat furniture. It looks a lot better than just having plain old desks.
> —Damien Jaipaul

> I like computers because they help you work faster and it is a lot neater. I like computers because your hands don't hurt as much as when you use a pencil.
> —Emma Braim

What happens in a classroom where students have continual access to computers and truly integrate the computer into the curriculum? The purpose of the project was not to provide another school computer lab, but to provide a working model of a computer integrated classroom. Malmo ACI develops, tests and disseminates computer based curriculum materials and encourages research.

Furniture, Hardware, and Software

> I like my computer furniture a lot because I can just move my chair over. I think it is really neat that we can just pull out our keyboards to do our work.
> —Laura Stewart
>
> I like the furniture because I just have to move over to work on the computer. There are nice places to put the disk drives and disks.
> —Jacinda Duquette

Furniture especially appropriate for cooperative learning activities and designed to accommodate young students was built. Seven octagon-shaped tables, each with four wings, were constructed in three sizes. The central table provides book storage space and a work area for students. A wing to the left of each student houses the computer, the disk drives, and a pull out shelf for the keyboard. Electrical and network cords are all concealed within the furniture. Only the monitor is located on top, making the student the focus of the classroom.

Prior to the introduction of the project, Malmo was a school of under 200 students with 6 computers and a printer. Now as we begin our fourth year of the project, Malmo has 300 in-school students, 67 computers (15 of these are Macintosh), six printers (two are Laser printers), a local area network , two modems, and a very active, integrated computer program. Two grade 3/4 ACI classes now share a full class set of Apple IIgs computers, an ImageWriter II printer, a LaserWriter IINT printer, and a file server using *AppleShare* software. The third grade 3/4 class shares a half class set of hardware. *AppleWorks* software, provided by ACEF, is our most frequently used piece of software. We have plans to set up a Macintosh computer classroom for grade 5/6 students in the near future.

Making the Computer an Effective Tool

It is hard to imagine trying to use the computer as a writing tool without the ability to touch type. Young students (grades 2 and 3) do not question their ability to learn keyboarding and research shows they can be successful. Some teachers and administrators question the feasibility of teaching touch typewriting because of the size of the hands of young students but we have found this not to be a problem.

The teacher is the key to success. The attitude of the teacher is a strong determining factor in the success of young students learning to touch type. In addition to an informed, enthusiastic, understanding teacher, one needs access to computer equipment, appropriate furniture, a fairly large block of time, a keyboarding program designed for the young student, and a purpose for learning.

The Keyboarding Program

> Keyboarding is easier than going one finger at a time and you don't have to sharpen your pencil. I can type better than my dad.
> —Jacinda Duquette
>
> I am glad that we had keyboarding because if we didn't I probably wouldn't be this good. I still make mistakes sometimes but not some of the stupid mistakes I used to make.
> —Naomi Mathison

A keyboarding program based on teacher directed lessons, not a keyboarding software package, is used in the ACI. The students work within the word processing package (*AppleWorks* in our case, although any word processor would be fine) while the teacher instructs and encourages. A text for keyboarding has been designed for our young students. Larger print, shorter line length, and reasonable spacing of lines, all for easier visual tracking, are some of the ways this material differs from the conventional typewriting manuals. A focus on language arts concepts and a smooth transition from keyboarding to composing at the computer is also included.

Students were instructed in keyboarding for 30 minutes a day, at least four days a week. Emphasis on technique, including correct fingering, proper striking, correct posture, and reading from the copy is evident during the presentation of the keyboard. Except for frequent and cumulative review, lessons included the introduction of two new keys, and stressed technique and speed. Instruction continued for approximately 50 lessons, and during the remainder of the year occasional review lessons were held. By the end of one year, these eight and nine year-old students averaged 20-30 words per minute (wpm) with a few students consistently over 40 wpm. (Pencil and paper scores were 8-10 wpm.)

Using the Computer as a Writing Tool

> I like storywriting because we can use quotation marks. Also the computer is a faster way to write. I like writing paragraphs. I also like writing whatever length of paragraphs I want!
> —Naomi Mathison
>
> I like story writing. I don't stop to think what I want to write on the computer. I just think what I want while I'm typing. I like the printer because you can have your things you wrote printed on paper. The printer is nice

> because you can take the things that you did well. You can take it home and have it on your computer, too.
> —Lisa Roan

Once the alphabetic keys had been introduced, including the shift keys, and children had gained some fluency, they were encouraged to use the computer as a means of completing their language arts assignments. Six to eight weeks were required before students were ready to make more extensive use of the computer. An early activity was to work with data bases, where students were able to enter information about themselves with a minimum of keyboarding.

Soon students began to compose stories, poems, and letters at the computer. Techniques for word processing were introduced gradually. Saving and printing routines had been introduced in the keyboarding sessions. Students were shown how to center titles, double space, and use the tab key during some of the later keyboarding lessons, and put these into practice when they were ready to print a story. Other features such as bold and underline were introduced as needed.

The computer was used extensively for written responses associated with our reading activities. Students responded on the computer to short stories, theme units, and our novel study units. Class story books and individual student books were written, edited, and printed on the computer. They were sometimes illustrated using computer graphics. Poems for Christmas, Valentine's Day, spring, and other themes, were written as a part of our program. Students used editing features of the computer and enjoyed the special effects of increased font size, bolding, and shaping their work.

We found students tended to be more willing writers when using the computer. They wrote longer stories, and did more editing. Editing was done both on the screen and on hard copy, with assistance from peers, parents, teachers, student teachers, volunteers, and senior citizens. Student work was displayed in the classroom, coiled in booklets and added to the library collection, shared with their friends and family, and included in newsletters and students' publications.

> The printer is a lot neater than my writing, I must admit. The sound bugs my ears a lot. Writing stories is easier on the computer. The writing goes faster but not the thinking. My stories are getting better.
> —Kerry Precht

> I'm glad that to print our files we just have to press a couple of keys and it just takes about 15 seconds, when last year it took a couple of days to get it down on paper. I like to have my own computer because I don't have to do half of it (my work) with pencil and paper. I like to do some things with other people like making pictures with the computer.
> I like writing both poems and stories on the computer. I think that it is always easier to write stories on the computer.
> I think I can fit in quotation marks easier on the computer. Paragraphs are easy, also. It is hard sometimes but it is always fun for me.
> —Laura Stewart

The opportunities that we have had to integrate the computer into the curriculum have allowed us to help students prepare for this age of information. It only makes sense that the students we teach use and become proficient with the technology of the age.

Telecommunications

The excitement and advantages of telecommunication

Since May of 1989, an added feature in our program has been our involvement in the Apple Global Education (AGE) Network. Telecommunications has opened a whole new area of interest and learning for the students at Malmo, and made maps and messages very important. "May I send a link to . . . ?" or "Have I got a message back yet?", are frequently asked questions by our students.

We are using Macintosh computers, *AppleLink* software, and a Practical Peripherals modem. Links are written directly in *AppleLink*, or on the Apple IIgs and transferred to the Mac. They can then be sent as a disk file or further converted to *AppleLink*. Both electronic mail and an extensive bulletin board system are involved. Text, graphics, and sound are transportable. A number of *HyperCard* stacks have been linked. Files written in most any Macintosh software can be sent as disk files. Graphics in *MacDraw*, *MacPaint*, or *SuperPaint* can also be sent.

Apple Global Education Network now has approximately 150 schools participating. These schools are located in Canada, United States, most of the European countries, Australia, New Zealand, Indonesia, Japan, Chile, Israel, Bahrain, and Saudi Arabia. A school in Russia has also been identified and soon we expect to communicate with them. All of the schools must be able to communicate in English, but English is not the only language of communication, as messages come in French, German, Spanish, Finnish, and Latin. Hopefully, Arabic, Inuktitut, and other language fonts will also be available soon.

The avenue of telecommunications provides our students with a real audience to receive their writing and students work hard to prepare the messages they send. They are especially concerned about the quality of their work when they are linking to students who are learning English as a second language.

Objectives

We have set a number of student objectives in relation to our telecommunications program. We are making changes in our existing curriculum to accommodate the very special opportunities that telecommunications makes possible. We are also taking every opportunity to use telecommunications in the existing curriculum and find this to be an enriching experience.

Student objectives:

- Students will be provided with an exciting opportunity for receptive and expressive language skills,
- Students will acquire, evaluate, and use information and ideas,
- Students will gain a greater awareness of others, learning to respect the dignity and worth of others and self,
- Students will learn basic computer operation and terminology,
- Students will achieve a greater understanding of computer technology.

Elements in telecommunications

Technical Considerations. A computer, a modem, appropriate communications software, and a telephone line are required for telecommunication. More student involvement and ownership are the advantages of having equipment in the classroom or library area, rather than in the school office.

Keyboarding. The advantage of students having keyboarding skills soon becomes evident. Our grade three through six students had numerous occasions to enter links and receive responses. They are able to sit down at the computer, concentrate on the message they are sending, and quickly key in the text they desire. Students without keyboarding skills become tired using the hunt and peck approach and often do not provide a complete response. Lack of keyboarding skills was mentioned by others on our network as one of the biggest drawbacks to student participation in the area of telecommunications.

Social considerations. Involving students in telecommunications provides them with the opportunity to gather information via the network and interact with students and/or teachers in other parts of our country and the world. They receive up-to-date answers from real people, often children their own age. This adds a dimension that does not exist in the books of the school library and is very motivational!

At times it is necessary to deal with language differences and stereotypes. Due to the nature of our school population, we have a focus on multiculturalism, so our network connections are used to add significantly to this aspect. We found our global communications network a great source of information and support for another focus, global environmental concerns, which are shared by many students around the world!

Monitoring considerations. Monitoring by a teacher or group of teachers is very important in order to keep the caliber of the projects at a suitable level. Unsupervised penpal activities soon degenerate into less than desirable drivel. Our students and teachers have proved to be a very responsible group. Ground rules for writing projects, well defined procedures for all projects, and continual teacher participation and supervision are important if projects are to succeed.

Timeline considerations. Setting timelines on projects and tying them into the school calendar seems to make a project more workable. Projects that run for the fall, winter, or spring term work well. Since we are a world wide project this is not strictly adhered to due to the variety of agendas in other parts of the world. Not everyone operates on the same calendar as our Canadian schools! We have enjoyed the spontaneousness of the network and tried to be flexible in our programming.

Curriculum considerations. Relating telecommunications projects to on-going curriculum activities within the school does make the projects more meaningful than if they were done in isolation. In order to accommodate others on the network, one does become involved in projects which would likely not have been a part of the regular school program. We have found many of these projects to be very worthwhile. If there is a class or a group of students that are willing to take on a project, and the project shows potential for growth and understanding, we will try to become involved. A project proposed by an Austrian school on the status of seniors would not have been a part of our program, but efforts to accommodate their project proved to be a very worthwhile learning experience for our students. There are times when the response to someone's project comes from a teacher rather than students. Not always do we have a group of students ready to be involved in every project, and yet the input of data from our area (we are the only Alberta school on the AGE network) would aid the project.

Each project is accompanied by a number of off-line experiences which form a very important, and much more time consuming part of the activity. Pre- and post-activities may involve individual students, small groups, whole classes of students, or sometimes the whole school.

Successful telecommunications projects

In order to be successful, the provision of some **structure** to a telecommunications project is an important element . It is necessary to set specific goals, expect specific tasks to be carried out, and anticipate specific outcomes. Having one person or a small group of people take **ownership of a project** works effectively. Then the project will be coordinated, from the call for collaborators to the summary and distribution of results.

Setting a **time frame** is also important. It is necessary to provide some lead time. This allows others to consider the project, decide who will be involved, and gather the necessary resources. Projects that have a set beginning and ending time are more likely to be successful. A deadline for response submission is necessary if data is being collected by various sites, so that final consolidation of all information can be carried out and returned to participating sites.

Small scale testing before launching the main project seems to be a good idea, but does not happen often. One thing that does happen is the repeat of a project that was especially successful. This sometimes includes the same schools as well as new ones.

The **format** in which data is recorded is one of the aspects that needs careful consideration. If the format is not clear it causes uncertainty among those participating and makes it more difficult for hosting students to accurately and effectively use the data that comes in.

What happens on the network

Although there is definitely a place for penpals and one time links, the most interesting and the greatest educational value has been produced by the more extensive projects. Use of this exciting means of gathering first hand information for units in health, science, and social studies, as well as enriching our language arts program, has motivated us to be as active as possible.

Christmas projects. In December of 1989, when my class was learning about Christmas customs of various people, we placed an outline on our computer bulletin board asking for information from the schools, and received much valuable information, all structured within our outline. We were pleased to have high school students as well as elementary school students respond. An added touch was provided by *HyperCard* stacks containing Christmas graphics and songs. A link from Helsinki caused the biggest stir. A group of high school students explained their Christmas customs and had the nerve to claim ownership of Santa Claus, saying that Santa lived in Lapland, their country. My students were appalled and declared, "No, he's ours. He lives at the North Pole." We got out the globe and discovered that Helsinki is much closer to the North Pole than Edmonton. "But he's still ours!"

Our 1990 Christmas project was one that swept our whole school and many others on our network. Since Malmo was in charge of our daily AGE$ News, we passed on to other schools the highlights we received. In addition to the usual information on Christmas celebrations, we received from Kangaroo Flat Primary School, in Bendigo, Australia, their own version of The Twelve Days of Christmas. Soon we had many versions to sing in our school assemblies! Can you tell where these lines originated?

- 12 Navajos weaving, 11 lobos leaping, 10 bobcats bouncing, 9 coyotes calling, 8 prickly pears a-pricklin', 7 senoras singing, 6 rattlers rattling, 5 golden zias, 4 ristras of chili, 3 clay post, 2 enchiladas, and a roadrunner in a yucca bush.

- 12 boreal forests, 11 ranchers roaming, 10 wild roses wilting, 9 glaciers glowing, 7 buffalo snorting, 6 Brownings quading, 5 beaver dams, 4 freezing falcons, 3 winsome wapitis, 2 gushing oilwells, and a horned owl in a fir tree.

- 12 skyscrapers, 11 Broadway shows, 10 traffic jams, 9 blocked bridges, 8 "Giants" linesman, 7 dirty playgrounds, 6 bratty children, 5 burger kings, 4 G.I. Joes, 3 supermarkets, 2 subway tokens, and a pigeon dumping on me.

- 12 parrots prattling, 11 numbats nagging, 10 lizards leaping, 9 wambats working, 8 possums playing, 7 climbing koalas, 6 platypuses, 5 kangaroos, 4 kookaburras, 3 jabirus, 2 pink galahs, and an emu up a gum tree.

Getting to know our network neighbors. For the first 26 school days of 1991, we have sent and received information from several sites around the globe, focussing on one letter of the alphabet each day. In each message we tried to share something important about our location that would be of interest to the rest of the participating schools. At Malmo School these messages became the contents of a huge wall display, and were turned into a school wide contest.

A What would you do with an amoutiq?
Name a Spanish mission that was built in 1718
and used as a fort?
Name the largest monolith that is situated on
traditional sacred aboriginal land.
How did Alberta get its name?

Answers to these questions were found by searching the A column of our wall data base.

Folk literature. A project on folk literature from the various areas of our global network has been of special interest to us. Here again the topic we thought would be of special interest to elementary school students has also been well received by high school students. One group of high school honors English students took on the project of writing original, modern fairy tales to share with everyone. Many versions of the Cinderella story, versions of other common folktales as well as Inuit, Indian, and Metis folktales were submitted. Graphics accompanied some of these folktales as well.

Social studies units. Our grade three students have been able to link to the schools on Baffin Island while doing a social studies unit on the Inuits of Northern Canada. Questions that could not be answered by using our books were answered by Inuit students and their teachers on Baffin Island. Being able to make direct contact with the Inuit children has helped our students to increase their understanding of our native people.

Grade five students linked to other classes across Canada to gain information on Canadian lifestyles. They compared the information received to their own lifestyle. As links go back and forth, friendships are built and information is expanded and refined. We focus on similarities.

A two-month study of Australia by Malmo grade 3/4 students was enriched by the many links that came to us from our primary school in Bendigo. Sea mail packages also were sent both ways across the ocean.

Current affairs. Involvement in matters of current affairs is a natural telecommunications activity. Our students are responsible for a daily newspaper called This Day in AGE$. The content of the news items ranges from international to in-school. Involvement in the publication of newspapers has taken many forms. Students who participate benefit from the activity, whether the papers were long or short, general or topical, elementary or secondary.

The excitement of first hand messages from California at the time of the big earthquake resulted in bulletin board displays and copies of links being taken home by students. We learned about things over the network that we never heard on television or radio. The *Earthquake HyperCard* stack produced by Apple has been of special interest to our students.

Currently there is much information on the Berlin Wall and the Gulf Crisis. Older students are asking questions and stating opinions. The effect of the media is a topic of interest. Students in different parts of the world are comparing the coverage they receive. Since we have an Arabic bilingual program in our school, the issues of the Middle East have been approached by students as young as grade three. Issues of multiculturalism and discrimination are important.

By linking messages to Vienna we were able to take part in Austria's computer fair. We kept in touch with Polar explorers through the news sent from our schools in Norway. We have followed the 1991 Alpine Ski Championships in Austria in a very special way. Students and teachers there sent many messages to the rest of the world, not to give us the information already in our newspapers, but to add those items that were of special interest to our students. Their special assignment was to go out and interview police and security people, hotel and hospitality personal, the environmentalists, and the skiers themselves. Then they reported to the rest of us on the network. In addition to their special assignment of interviewing, the students in our AGE school in Saalbach took special telecommunications greetings from Malmo students to Edmonton's Edi Podivinsky. Edi was thrilled. They even cheered for OUR skier!

Health, science, and environmental studies. Various mini projects on care of the elderly, food and meals, and drug abuse have been explored.

Daily weather reports were linked during the month of February 1990, from a number of sites to a school in Nova Scotia. They then returned charts and graphs to the participants. Information on acid rain has been collected and linked to others on the network. During the 1990-91 school year our weather collections have consisted of a weeks recordings every five weeks. This way we get a glimpse of what it is like in each site throughout the year.

Articles in the newspapers are often a source for a mini project, such as the article about the Canadian government discussing the matter of (making a stink about) disposable diapers. A grade four student linked out a response to this to see if others had similar concerns and found that an article in one of our Canadian magazines had been read by a respondent in Saudi Arabia.

Mathematics and art. Surveys conducted at various grade levels, often serve as graphing material for younger students, while older ones use it in spreadsheets and graphs. The information has been shared in many formats. Problem solving activities have also been linked.

Using various graphics programs, students have looked at architecture and how buildings vary in different locations. Young students (grades 1-3) made drawings of their homes, which then became the blocks for a composite "quilt". Having blocks from Norway and New Mexico in a quilt certainly made it unique. Illustrations to go with stories have been linked. Other projects included an activity on fingerprints, and one called Imagine, where students drew and then wrote about the five most important words they would share with an alien.

Language arts. What better way is there to motivate letter writing than to have children write letters, link them on the network, and then receive answers promptly? A vast improvement in the letter writing skills was noted as grade three students responded with 200-300 word letters.

An interactive poetry project involving the writing of couplets—two line rhyming poems—on the theme of winter versus spring, was started by my grade three students. After linking only the first line of these couplets to a number of schools and were delighted with the results. Students in Spain and Norway (ages 13 to 16), learning English as a second language responded with their second line. Inuit children on Baffin Island as well as children in Massachusetts and Arizona sent responses.

A story about a green monster who ate broccoli was sent to us on Saint Patrick's Day from students at Pond Inlet on Baffin Island. In a sequel to this story, the monster visited our city of Edmonton and Malmo School. Once my students had had their fun, they sent him (links, of course) back to Pond Inlet.

Students are currently involved in an Online Writers' Workshop. Students and teachers post biographical information. Students can then post a story that is nearly ready for publication. They are also expected to respond to the stories of others. It is our objective that every posted story will have at least one response within two days.

Our language arts program has been enriched by these and numerous other writing projects which have provided opportunities for our students to write using computers.

One of the exciting things about network communication has been the activity between students in different age groups. There has also been a rich dialogue between administrators, teachers, student teachers, and students. Helsinki high school students are special to Malmo grade three students.

Conclusions

Our success in the keyboarding program at the grade three level has been very rewarding. Students become efficient computer users. The computer has become a powerful tool in language arts and other areas, and students have shown a marked growth in their writing skills.

AGE telecommunications has provided many benefits for both students and teachers. This new and exciting way to exchange information has provided students with a real audience for their writing. Students quickly gain first hand information from students in other parts of the world. They share their concerns for a safe global environment, for peace, and freedom, and gain an understanding for the feelings of others.

Teachers consult and compare with other teachers in our own country, continent and around the world, sharing projects, problems, and solutions. We trust that we can improve global education and make the world a better place for future generations as we work together through AGE.

Paper
Learning Circle Projects Sponsored by Global Partners

Margaret Riel
Adel Labib
AT&T Learning Network

Abstract

This paper describes cooperative learning across classrooms. Students from distant locations are linked by computer to a new educational environment called a Learning Circle. Like cooperative learning approaches in the classroom, students working across classrooms have the opportunity to learn from and teach one another. But the increased diversity of perspectives represented in a group of classrooms and the written mode of communication provides new ways to develop students' cognitive and social skills. Electronic Learning Circles provide a mechanism for developing cooperative learning among students in distant locations and at the same time provide a vehicle for the professional development of teachers while they remain in their classrooms working with students.

> Cooperative learning is not a new idea—it is as old as humankind. The capacity to work cooperatively has been a major contributor to the survival of our species. Throughout human history, it has been those individuals who could organize and coordinate their efforts to achieve a common purpose that have been most successful in virtually any human endeavor (Johnson, Johnson, Holubec & Roy, 1984: 13).

Success in the information age will depend on our ability to use new forms of technology and communication to facilitate our collective understanding of the forces that shape our world. Time and space are no longer significant barriers to collaborative work. At the present time, the rapid pace of technological development outstrips our speed in developing the social and intellectual skills necessary for using new technology in creative and innovative ways. New forms of electronic communication will necessitate new forms of cooperative work. We do not inherit social skills; their development requires modeling, repetition, experimentation and continual feedback. This skill development begins in the home and needs to be continued in the school. When students work primarily in isolation, they fail to learn some of the most valuable lessons that school has to offer.

School experience, Dewey argued at the turn of the century, should weave into its very fabric the values, social order, and processes it seeks to impart and not merely teach about these processes. Likewise, students do not need to **hear** about how technology can change the world. Nor should schools focus on narrow approaches to computer literacy which teach about computer mechanics. What schools need to do is help students develop the social and intellectual skills necessary to use these machines to their potential. These skills are acquired through experience and not lectures. Cooperative learning principles point the way to develop an instructional system that can help children assume their role as responsible adults prepared to work with one another to shape their collective destiny in a shrinking world.

Cooperative learning provides a mechanism for developing students' school experience in the spirit of Dewey's fundamental principle of direct experience. Cognitive psychologists have recently characterized cognitive development as an inherently social and cultural process (Cole, 1981; Norman, 1980; Lave, 1988; Wertsch, 1985). Research on cooperative learning strategies has shown the effectiveness of this approach in learning. A review of over 100 research studies compared the benefits of cooperative learning with individualistic or competitive instructional approaches.

> Results indicate that cooperative learning experiences tend to promote higher achievement than do competitive and individualistic learning experiences. These results hold for all age levels, for all subject areas, and for tasks involving concept attainment, verbal problem solving, categorization, spatial problem solving, retention and memory, motor performance and guessing-judging-predicting. For rote-decoding and correcting tasks, cooperation seems to be equally effective as competitive and individualistic learning procedures (Johnson, Maruyama, Johnson, Nelson & Skon, 1981).

Cooperative learning strategies have been found to increase academic skills, develop social skills, aid mainstreaming of handicapped students, reduce ethnic tensions and increase self-esteem among school children (Slavin, 1983; Sharan, 1984; Kagan, 1986). When teachers shift classroom lessons from whole group instruction to small group investigations or team projects there is an improvement in instruction and learning which fosters pro-social patterns of peer interactions and relationships.

Comparison of Cooperative Learning Within and Across Classrooms

Telecommunications offer schools the opportunity to extend cooperative learning across classrooms through computer telecommunications ("telecomputing"). Electronic networks make it possible to design educational programs that bring students and teachers into working relationships with each other. These across-classroom programs are very similar to cooperative learning programs designed for single classrooms. They can be structured to create positive interdependence on specific educational tasks with individual accountability. But the differences are also significant. The students must work cooperatively in the face-to-face situation of the classroom and work cooperatively in written communication across the network.

Cooperative Learning Within Classrooms: Group Investigations

Cooperative learning in the classroom involves organizing students into teams or groups. Sharan and Hertz-Lazarowitz (1979) have developed a procedure for organizing group investigations within classrooms. Successful organization of group learning requires the coordination of four dimensions of classroom life:

1. The organization of the classroom into a "group of groups." The teacher divides the class into a number of small groups or teams that will be working together on specific tasks. The size of the groups and the mix of the students are important variables.

2. Use of multifaceted learning tasks for cooperative group investigations. The teacher plans the academic task with a clear group goal. The goal should require the interdependence of the group but the individual contributions of all should be clear.

3. Inclusion of multilateral communication among students and encouragement of active learning skills. The students take an active part in different parts of the task and share the information to accomplish the group task.

4. Teacher exchange with each of the groups. The teacher needs to be available to help groups that need more direction or ideas to move forward, but teachers should not solve group problems. The role of the teacher is to help the groups solve their own problems.

In this model, a teacher forms the groups and identifies a specific task. Student teams work together to plan and carry out the task. They prepare a report to present to their class for discussion and evaluation. The teacher works with the individual teams giving them directions and the teams report back to the teacher and class as a whole. While people on the same team work closely together, there is little cross-team interaction. In some models of cooperative learning, the teams compete with one another to complete the task.

Cooperative Learning Across Classrooms: Learning Circles

The Learning Circle design of the AT&T Learning Network[1] parallels Sharan's group investigation model by applying similar procedures *across* classrooms. A Learning Circle is a small number of classrooms that interact electronically to accomplish a shared goal. Each classroom in a Learning Circle is a team that contributes to the overall end product.

Learning Circle interaction takes place over a four-month period and is organized around the following sequence of activities:

1. Forming the Learning Circle. The Learning Circle, a "group of groups", is formed by AT&T. Approximately eight geographically diverse classrooms with a common educational focus are joined together. The Learning Circle program provides the electronic link and structures introductory activities to help students and teachers learn more about the schools and communities of their partners.

2. Planning the Learning Circle Projects. Teachers and students from each of the classrooms plan a learning task for the Circle. They offer their plan to the Learning Circle for group comments. They, in turn, help shape the plans of the other classrooms (students/teacher teams). The work of each of the classrooms on the projects is clearly defined.

3. Exchanging Work on the Projects. Students work closely with students in their own classroom as well as the students in distant locations to carry out their learning activities. Students receive the guidance and encouragement of teachers in other locations and from the Learning Circle coordinator (teacher).

4. Creating the Publication- The Group Task. Each class of students is responsible for collecting, analyzing and arranging student work on the project they sponsored. The Learning Circle projects from each class are combined and assembled into a Learning Circle publication.

5. Evaluating the Process. Students and teachers reflect back on their time working together, read their Circle

---

[1] The AT&T Learning Network, Post Office Box 4012, Bridgewater, New Jersey 08807-4012, phone, 800 367-7225

publication, and evaluate the learning that has taken place on the network. These reflections close the Learning Circle.

Comparison of Group Investigations and Learning Circles

The sequence of activities in a Learning Circle is very similar to that of the classroom model of cooperative learning. However, there are two important ways in which Learning Circles differ from the group investigation model of cooperative learning. One is the role of the classroom teacher and the other is the interaction between the teams.

In cooperative group investigations in the classroom, the teacher forms the groups and defines the academic task. The teacher maintains complete control over the formation of the groups and the work that each of the groups will be expected to complete.

In this model of across classroom learning, the groups (individual classrooms) have been formed into a Learning Circle by AT&T and each group defines one of the projects for the Learning Circle. A teacher works closely with his or her students to plan their project. However, the other projects are planned by the distant teacher/class teams. This causes a shift in the power relationship in the classroom. The teacher joins the class as a learner and problem solver. One classroom teacher does not have total control over the direction of the Circle activity. Without advanced knowledge of the projects that students and teachers in other locations will contribute, teachers as well as students are challenged to explore new areas. The teachers' excitement over the discovery of new information demonstrates the high value they place on learning. It is often the teachers' enthusiasm that appears to motivate the students' learning.

The second difference concerns patterns of communication. Learning Circles encourage more communication *across* the teams (classrooms) than is commonly found in the cooperative learning model. Each classroom plans a project and assumes the responsibility of creating a final report or publication to share with the group. This is very similar to the group investigation program described by Sharan. But to carry out the project, the students request the help of other classes and teachers (teams) in their Learning Circle. In return for this help, they offer to help the other classrooms with their projects. So while each class (team) is working on a project they proposed, they are contributing to the work of the other classes in the Circle.

The structure of asynchronous communication on networks makes it possible for students to interact with a number of groups at the same time without creating the chaotic situation that would result if this were attempted in the classroom setting. Each of the 6-8 projects in a Learning Circle represents the collective work of all and is summarized in a report by the classroom that sponsored the project. This framework creates cooperative work within a team in a way that is similar to the cooperative learning framework, but it also constructs cooperative working patterns *between* the teams.

The Circle Coordinator facilitates the group organization, but, since each team (classroom) has a very skilled person (teacher) helping to organize their contribution, there is less of a need for the strong centralized control that often characterizes classroom programs. In this way, electronic networks provide a new way to organize cooperative learning *within* and *across* classrooms. It provides a tool that helps students and teachers work collaboratively with their peers in distant locations.

Cooperative Learning in Learning Circles

There have been a number of articles and books describing the potential influence of telecomputing in the classroom (Levin & Cohen, 1985; Newman, 1987; Harasim, 1990, Riel, 1989). Often these papers report on a very exciting exchange that took place between two or more classrooms. They do not, however, discuss the large number of projects that were planned but either never begun or ended in disappointment. A naive but common assumption is that networking is intrinsically motivating and if students are placed in contact with other classes, exciting projects will be the inevitable outcome. Unfortunately, many networking efforts end prematurely (Levin & Riel, 1990). Cooperation across classrooms takes careful planning, implementation and support. The AT&T Learning Network has spent a number of years researching how to provide the necessary support for a large number of classrooms working in Learning Circles. The resulting level of cooperative learning among sets of classes in Learning Circles has been quite impressive.

This paper describes the activity of all eleven elementary level Learning Circles during the spring of 1990 on the AT&T Learning Network (the middle and high school level Learning Circles will be described in subsequent papers). The elementary Learning Circles had four different subject area themes as can be seen in the chart below. Computer Chronicles Circles focused on journalism skills, Places and Perspectives Circles on social science skills, Mind Works Circles on writing skills, and Energy Works Circles on science skills. Using the sequence of activities of a Learning Circle described previously, the progress of each of the eleven Circles will be compared as they worked together as a group to accomplish their goals.

Elementary Learning Circles, Spring, 1990

Computer Chronicles:
Circle a: 8 classes
b: 7 classes
c: 7 classes
Mind Works:
Circle a: 7 classes
b: 8 classes
c: 7 classes

Places and Perspectives:
Circle a: 8 classes
b: 9 classes
c: 8 classes
d: 8 classes
Energy Works:
Circle a: 9 classes

Total all Circles: 86 classes

1. Forming the Learning Circle

The A&T Learning Network sets up the electronic connections for Learning Circles and sends all the software and curricular materials necessary for participation. During this period, the teachers were asked to respond to the electronic Roll Call. The teachers exchanged messages and students joined by completing and exchanging Classroom Surveys. During this two week period, 94% or 81 of the 86 classes responded to the electronic Roll Call. The high level of participation in the first weeks suggests that the AT&T communication software, automated sequences for message exchange, and access to toll-free customer support provided adequate training for teachers to learn how to use the technology. Two of the five missing classrooms responded to the Roll Call late, while the other three never made contact.

On the average, 5 of the 8 classes in a Learning Circle sent their Classroom Survey by the end of the second week. In most cases, the other classes did complete the Classroom Survey in subsequent weeks.

2. Planning the Learning Circle Projects

Once teachers and students had met the other members of their Learning Circle, they were ready to begin planning the Circle curriculum—a set of projects defined by the classes themselves. Each class had the opportunity and was encouraged to sponsor a Learning Circle project. However, it was also possible to be a participant in projects sponsored by others without sponsoring a project. A curriculum guide for the Circle described how to integrate their Circle project with their regular instructional plans. During the next two weeks, teachers and students offered their project plans to the Learning Circle for group comments. They, in turn, helped shape the plans of the other class/teacher teams. A Learning Circle project in a Computer Chronicles and Mind Works Circle would be to sponsor a section of a newspaper (sports, news, opinions, etc.) or of a writing journal (poetry, circle stories, biographies, etc.). The projects in a Places and Perspectives or Energy Works Circle often involved some research activities in addition to writing, such as computing average temperatures, collecting historical information, or experimenting with mechanics.

Most classes did sponsor one or more projects. In eight of the eleven Circles, everyone in the Circle participated by sending at least one message during this two week planning period. Seventy-one of the eighty-three teachers (86%) sponsored a Learning Circle project. This means that most of the teacher and student groups had the necessary support and direction to feel comfortable sponsoring a learning activity in this new environment.

3. Exchanging Work on the Projects

Students worked closely with students in their own classroom as well as with students in distant locations to carry out their learning activities. Learning Circle Coordinators helped monitor the Circle progress on each of the projects. Ideally, each of the Learning Circle projects should receive work from most of the other classes during this period. All 83 classes sent messages to the Learning Circle during this period.

In four of the Learning Circles the coordinators posted the responses by school, so it was possible to monitor the number of schools that sent student work on each of the projects. The average number of classes participating in each project in these Circles was 6.4 classes. This high level of participation shows that classes took their responsibility to the group seriously. Students were motivated to do their assigned work because they knew that their peers in distant locations depended on them. If reminders were sent to those who had not completed their work, they were more likely to have been sent by students themselves rather than by teachers.

4. Creating the Publication

Each class in the Learning Circle collects, analyzes, and arranges the materials that were exchanged on their sponsored project into a printed format. This work forms one of the sections of the Learning Circle publication. Each class works as a team of editors to evaluate, select and edit the work to include in their section. This means that while a class may have contributed work, the student editors make the decision to include it or not to include it in the printed copy. Each class sends a copy of their section to the other participants. The Learning Circle Coordinator also sends a cover page, an opening letter introducing the participants and a table of contents.

All eleven Learning Circles produced a Circle publication. The average number of pages for each of these Circle publications was 64 with an average of eight separate sections. In each section there was work representing an average of 5.6 schools. Most of these publications contained very impressive work for the grade levels represented.

5. Sharing the Publication and Evaluating the Process

Teachers report that students are very excited to see their work in a final published form. The booklets have been shared with parents, presented to the school board, put on display in the school office and catalogued in the library. The students who take part in this cross-classroom cooperation are very proud of their work. They know that their words are also being read by students, teachers, and adults in many different locations. Their work is being read for the content—the information that is contained about themselves, their school and their community— rather than to evaluate their skills. They are published authors and they share the sense of pride that an author feels when the first copy of the book arrives from the publisher.

If the Circle finishes its work on time, there is time during the last two weeks on-line to assess the group process and reflect on what they have learned. In many cases, the teachers and students in the Circles described in this paper were working hard at finishing their part of the publication. In some of the Circles, the sections were finished, mailed, and the publication was assembled. Students in these Circles used the final period to share comments on their work.

Here is an example of a Circle closing message. This one was sent by a class that participated in the Computer Chronicles Learning Circle 2:

> Dear Learning Network friends,
> This is the time when we have to say our good-byes. It has been a fun-filled learning experience for all of us in the computer lab at Stockbridge Middle School. If it hadn't been for the marvels of technology, we would have not met such friendly people from around our country and even to another country. We learned how much the same we all are even hundreds of miles away from each other. It was fun to learn about the school practices of our long-distance friends. We also learned that telecommunication process can be frustrating if there are technical problems. We are very anxious to see the final product, with the contributions of all of us. It is great to see your own name in print! Finally we learned that it can be fun to write articles, and that even the process of editing is not all that painful! Maybe some day, some of us will be writing for the Washington Post, the Wall Street Journal or the local daily papers. We hope that all of you have a wonderful and safe summer. Good luck in your future.
>
> Computer II and III
> with Mrs. Ginny Nightingale
> Stockbridge Middle School
> Stockbridge, MI 49285

Cooperative Learning for Teachers

Cooperative learning enthusiasts describe peers—students—as valuable educational resources who could be used to change education. But teachers are also valuable educational resources for one another. Teachers provide for one another the same level of intellectual exchange as cooperative learning groups do for students. Consider this quote from Sarason:

> The assumption that teachers can create and maintain the conditions that make school learning and school living stimulating for children without the same conditions for teachers, has no warrant in the history of man (Sarason, 1976:123-124).

Implementation plans for cooperative learning describe as an important component professional support groups for teachers. In Learning Circles, the teachers in the Circle are the professional support group. Teachers' benefits can be described in four areas (Riel, 1990):

1. They increased their content knowledge by learning from students and teachers at a distance,
2. They developed new teaching and learning strategies directly from their work on the network and from their contact with other teachers,
3. They often experienced increased self-esteem as other teachers complimented them on their projects and student work,
4. They developed national and international professional and personal relationships.

Conclusions

Schools have in the past failed to realize the strong force that social interaction plays in the development of intellectual skills. With the increased recognition of the role of social factors on cognitive development in the social sciences (Norman, 1980; Cole, 1981; Lave, 1988; Wertsch, 1985), educators are becoming more aware of the importance of social interaction in classrooms. Cooperative learning strategies are powerful techniques for developing social and intellectual skills for peers and are a valuable educational resource for improving classroom learning.

The development of cooperative learning across classrooms on networks is guided by the same issues that define learning within classrooms. As this brief summary of AT&T Learning Circle shows, with telecomputing it is also possible to use the principles of cooperative learning to create a "group of groups" across classrooms. Electronic networking makes it possible to create highly interactive groups of students and teachers who remain physically and temporally separated. The power of electronic networks to facilitate *group* communication *among* classrooms and teachers is opening up the classroom in ways that Dewey never imagined. Classroom telecomputing can narrow the gap between direct experiences and formal instruction.

But teachers cannot be expected to reorganize classrooms to provide this type of experience for students while they remain isolated in the traditional classroom. Electronic communication enables teachers to form new working relationships with educators throughout the world. Teachers and students can be linked across classrooms, working in teams to help solve real problems, to share cultural perspectives, and to learn from one another. This type of cooperative learning across classrooms can change the life of teachers and with it their methods for educating students. When teachers working on educational networks are asked to rank the benefits of educational networking, a consistently strong finding is that they rate their own learning, not the learning by their students, as the most important benefit of the program. In doing so, they give empirical evidence to Sarason's claim that educational change cannot bypass teachers.

The work described in this paper results from the cooperative work of students and teachers in distant locations. This is only the beginning of the new working relationships that are possible. Networking can also provide an avenue for different sectors of the community to reach out to classrooms. Learning Circles need to be extended to include other adults who are not currently part of the educational cycle. Bankers, newscasters, doctors, builders, manufacturers, scientists, reporters, politicians, and many more people could find it productive to develop working relationships with students who remain in classrooms. In some cases these adults might join in cooperative projects designed by students adding their expertise and guidance. But just as often, students could be engaged in work that will be of interest and value to the adults in their community. Computer networks allow students to serve as members of "tele-task forces" to address many of the issues that face adults every day.

Students can collect data, conduct interviews, give surveys, analyze numbers, and create materials that can significantly contribute to the adult community. Tele-task forces that include young adults still in school could begin to return to this segment of our society some of the self-respect that has been lost in their isolation in schools. They could once again be part of the productive society but remain in the protective setting of the classroom. Teachers would not be the only educators in the classroom, they would be joined by visiting educators from all sectors of society. Students and teachers would no longer be completely isolated from the adult community and our understanding of the term public education could be significantly altered.

But these are visions of the future. At present, we have only begun to explore the social realm that can be created with telecomputing. As students realize that they can create and use new information, information that becomes a part of our knowledge base, they will create the future.

References

Cole, M. (1981). *Society, mind and development, and the zone of proximal development: Where culture and cognition create each other.* (CHIP 106) La Jolla, California: Center for Human Information Processing, University of California, San Diego.

Dewey, J. (1916). *Democracy and Education.* New York: The Free Press.

Harasim, L. (1990). (Ed.) Online Education: Perspectives on a New Environment. New York: Praeger.

Johnson, D. W., Johnson, R. T., Holubec, E. J. & Roy, P. (1988). Circles of Learning: Cooperation in the Classroom. Alexandria, VA: Association for the Supervision of Curriculum.

Johnson, D. W., Maruyama, G., Johnson, R., Nelson, R. & Skon, L. (1981). Effects of cooperative, competitive and individualistic goal structures on achievement: A meta-analysis. *Psychological Bulletin*, **89**, 47-62.

Kagan, S. (1986). Cooperative learning and sociocultural factors in schooling. In *Beyond Language: Social and Cultural Factors in Schooling Language Minority Students.* Los Angeles: Evaluation, Dissemination and Assessment Center, California State University, Los Angeles.

Lave, J. (1988). *Cognition in practice.* New York: Cambridge University Press.

Levin, J. & Cohen, M. (1985). The world as an international science laboratory: Electronic networks for science instruction and problem solving. *Journal of Computers in Mathematics and Science Teaching*, **4**, 33-35.

Levin, J. A., & Riel, M. M. (1990). Building electronic communities: Successes and failures in computer networking. *Instructional Science*, **19**, 145-149.

Newman, D. (1987). Local and long distance computer networking for science classrooms. *Educational Technology*, **27**, 20-23.

Norman, D. (1980). Twelve Issues for Cognitive Science, *Cognitive Science*, **4**, 1-32.

Riel, M. M. (1985). The computer chronicles newswire: A functional learning environment for acquiring literacy skills. *Journal of Educational Computing Research*, **1**, 317-337.

Riel, M. (1989). Four models of educational telecommunications: Connections to the future. *Education and Computing*, 5, 261-274.

Riel, M. (1990). Cooperative learning across classrooms. *Instructional Science*, (in press).

Sarason, S. (1976). *The creation of settings and future societies.* San Francisco: Jossey-Bass.

Sharan, S. & Hertz-Lazarowitz, R. (1979). A group-investigation method of cooperative learning in the classroom. In S. Sharan, P. Hare, C. Webb & R. Hertz-Lazarowitz (Eds.), *Cooperation in Education.* Provo, Utah: Brigham Young University Press.

Sharan, S. (1984). *Cooperative learning in the classroom: Research in desegregated schools.* New Jersey: Lawrence Erlbaum Associates.

Slavin, R. (1983). *Cooperative Learning.* New York: Longman.

Wertsch, J. V. (1985). *Vygotsky and the social formation of mind.* Cambridge, MA: Harvard University Press.

Special Presentation—ACM/SIGCUE

Expanding the Base of Education Programs: Computer Based Training in the Corporate Environment

Regan Carey, Doris Carey
University of Colorado at Colorado Springs

This presentation will focus on how education programs can provide Computer Based Training (CBT) for the business world. Education programs will be highlighted that provide coursework for corporate clients.

Special Presentation

Global Mind Change— The Challenge of Restructuring Education

David Thornburg
Thornburg Center for Professional Development
San Carlos, California

More than a decade has elapsed since their introduction, yet personal computers have failed to be taken seriously as educational tools in many classrooms. Cost is no longer the major factor - attitudes must shift to bring technology to our students. American education is failing to keep up with the changes taking place in society as a whole. Unless we restructure how we teach, we risk graduating students who are ill-equipped to thrive in the 21st Century. This presentation focuses on public resistance to educational restructuring, the reasons that technology has barely entered our classrooms, and presents some concrete ideas that we can spply to help parents and others understand the importance of preparing children for their future, not for our past.

Special Presentation

Technology-Based Manipulative Approaches to Teaching Mathematics

Fred Ventura
Ventura Educational Systems
Newbury Park, California

New standards have set the course for a new direction in teaching mathematics in response to a recognized need to reform the mathematics curriculum. The content and emphasis of the elementary mathematics curriculum is changing. The traditional dominance of computation and other traditional skills had failed to foster the development of mathematical insight, logical reasoning, and problem solving skills and in contrast has been based on rote activities. Children loose interest in mathematics because it has been presented as a set of rules and procedures and they have not had the opportunity to actively participate in creating mathematical knowledge.

Special Presentation—ACM/SIGUCCS

NSFNET—A Network for Research and Education

Douglas Gale
Program Director, NSFNET
National Science Foundation
Washington, DC

This session will describe the NSFNET Program of the National Science Foundation and plans for the NREN. Uses of the network in predominantly undergraduate institutions will be presented. Current uses of NSFNET in the pre-college environment will be presented as well as ideas for the future use of the NREN in pre-college education.

Projects
Diagnostic Skills Trouble Shooting

Troubleshooting Complex Aircraft Systems: A Hypermedia CD-ROM Development Project

Martha H. Weller, Michael R. Michela[1]
University of Illinois

Abstract

A multimedia, computer-based instructional system is being developed jointly by faculty in the Pilot Training Department and the Aviation Research Laboratory of the Institute of Aviation (University of Illinois). The system will assist advanced aviation students (private pilots) in learning to troubleshoot electrical problems in small aircraft.

The full system will consist of a number of linked modules, including tutorials, simulations, scenario item banks, glossaries, and performance evaluations. The target delivery system is a Macintosh IICX with high resolution color graphics.

SuperCard (v. 1.5) serves as the core programming environment. Animations are being developed using MacroMind Director (v. 2.0). Several software packages are being used to manipulate graphics created by digitizing a variety of video sources (e.g., photographs, paper-based artwork, selected frames from video tape and video disc). Audio narration is digitized using Farallon's MacRecorder.

A brief videotape showing the current status of the project will be shown and the presenters will discuss technical and design issues involved in creating a massive multimedia project, including the pros and cons of CD-ROM as a delivery medium.

[1] Full project team also includes: Drs. Terry Harshbarger and Jeff Koonce, University of Illinois.

Computer Based Error Correction Simulations for Training English Composition Teachers

Lynne Diaz-Rico
California State University, San Bernardino

Abstract

Abstract unavailable at press time.

Knowledge-Based Inference Tool for Enhancing Clinical Diagnostic Skills

Frank J. Papa
Texas College of Osteopathic Medicine

Jon I. Young
Gerald A. Knezek
University of North Texas

J Shores
Texas College of Osteopathic Medicine

Abstract

This paper describes efforts to differentiate between expert and novice knowledge in medical diagnosis. The purpose of this differentiation is twofold: 1) to create a valid and reliable assessment instrument that could be efficiently administered to measure diagnostic ability; 2) to create the basis of an intelligent tutoring system that could be used as a more effective teaching vehicle.

Both activities depend on a computer system that receives feature frequency information on diseases in a specific medical problem space. This information is then transformed to determine a prototype of each disease state. Assessment of proficiency is based on how well each individual's prototype correctly diagnoses presenting diseases and differentiates between the correct diagnosis and the next most logical disease state based on available information.

Instruction is built in a hierarchical strategy that moves the student from a complete instructional sequence to a self-determined level of help that they system can be programmed to offer.

Special Presentation
Logo Action Research Collaborative's National Sites Report

The Logo Action Research Collaborative is a national teacher enhancement network funded by the National Science Foundation. It supports Logo teachers to take on new professional roles, conducting action research projects and developing assessment tools related to Logo learning. Teacher researchers in pilot projects have conducted case studies of individual students' learning and investigated a broad range of research questions relevant to Logo learning environments. These have included: mathematical topics such as variables, recursion and the role of 360 degrees; pedagogical topics such as cooperative learning, facilitating sharing meetings, designing intervention strategies, involving students in assessing their own work, and using contracts to support pairs of students in resolving conflicts; and computer science topics such as structured programming, effective use of procedures and sub-procedures, creating images to support disk management; and debugging.

This year the Logo Action Research Collaborative is supporting groups of teacher researchers at nine sites nationally. This will be the first report of the national dissemination of the project. The principal investigators will present the project overview and rationale, the Leadership Institute and support strategies, and the approach used for documentation and formative evaluation. Teachers and group leaders from several of the nine sites will present their experiences as first-time leaders and their preliminary findings about Logo learning and the professional development of teacher researchers. We will also share our experience of linking sites by telecommunication.

Panelists:

Daniel Watt, Molly Watt
Education Development Center
Newton, MA

Patricia Rowe
Boston, MA

Sheila Cory
Chapel Hill-Carrboro, NC

Phyllis Victorson
Chicago, IL

James MacNeil
Concord, MA

Marian Rosen
Ladue, MO

Special Presentation
Model Technology Schools Projects: What Have We Learned?

Abstract unavailable at press time.

Projects

Telecommunications in K-12 Environment

Telecommunications Across the Curriculum and Around the Globe

Brevard Williams III
Gwinnett County Public Schools, Georgia

Abstract

Do students in the Soviet Union waer car-rings? What do students in Washington, CT like to do after school? Why are Finnish students so fired-up about the Baltic Sea? Telecommunications opens up new possibilities in the are of global communications. Two projects, known as Scrapbook USA and Project: Bering-Bridge offer students and teachers the opportunity to communicate LIVE with their counterparts across the globe. This project focuses on developing dialog between students, helping students to identify and understand cultural similarities and differences and developing the writing process.

Established in early 1989, SCRAPBOOK USA is a collaborative effort between a Georgia teacher and technology coordinator, Bard Williams, and a high school English teacher from Washington, Connecticut, Ted Roth. Working together with personnel from Quantum Computer Service's America Online telecommunications service, Mr. Williams and Mr. Roth established a broad-based educators user-group dedicated to the exchange of cultural knowledge through the writing process. Since it's inception, the project has grown to include over 20 schools representing a variety of levels and cultural diversity. Recently, schools from the Soviet Union have also become involved.

Internet in the High School

Janet Meizel
Davis Unified School District, California

Abstract

Abstract unavailable at press time.

Creating Interdisciplinary Links Using Computer Teleconferencing

Peter Hutcher
GTE Educational Services
San Leandro, California

Abstract

Abstract unavailable at press time

Special Presentation
Arizona Educational Telecommunication Cooperative

A. Ashton
Arizona Board of Regents
Phoenix, Arizona

Abstract unavailable at press time.

Projects
Just Do It! Implementing Technology

How we Got to 'Yes' or Surmounting $1.6 Million in Hard Feelings

Richard Rose
Sweetwater School District #2
Green River, WY

Abstract

In 1990, the Sweetwater District #2 passed a bond issue for $1.6 million in Technology Expenditures. This generated about $6 million in faculty expectations. Most of these passions centered around acquiring hardware rather than meeting curricular objectives.

The answer was to place a one year freeze on all expenditures, which initially caused much frustration. But now, thanks to an orderly step-by-step process of shared responsibility and a little help from Fisher & Ury's *Getting to Yes: The Art of Negotiating Agreement Without Giving In*, we are well on our way to a comprehensive Mission-Based K-12 District Computer Plan that is owned by teachers, administrators, and community leaders.

The purpose of this presentation will be to discuss how we did it. Subtopics will include: Resistance to Planning (& How to overcome It), Getting Industry Involved, Planning Process Structure, Top-Down Planning Logic, and Healing the Legitimate Casualities of K-12 Planning.

Dr. Rose holds on Ed. D. in Curriculum and Instruction from the State University of NY (Albany) and an Ed. S. in Educational Management from New Mexico State Univerity. He is the VP and Newsletter Editor for the Wyoming Educational Computer Council and has spoken on Computer Planning to the Wyoming School Boards Association.

Solution: Project DEIKAS

Jerry Christy
Fort Benning Schools, Georgia

Abstract

"Sometimes you've just gotta' break the rules." is a current tag line used by a national fast-food chain and a strategy espoused by at least one well-known author of creativity development strategies. Moderate success in uniting teachers, students and computers prompted us to consider breaking a few rules we had in place if we were to realize significant benefits from the wealth of time and money invested in educational technology.

The Scene: School-wide and district-wide staff development courses "packed 'em in" year after year, but the enthusiasm generated in the classes seemed to dissipate once the teacher was responsible for delivery in the classroom. A small number of computer using teachers - motivated by personal curiosity and an adventurous spirit - were doing exciting things, but overall the implementation was disappointing.

Enter the Rule Breakers: Discussing the "less-than-ideal" state of affairs in the middle school one day over lunch, the school's media specialist and the district technology coordinator launched Project DEIKAS: Drag 'Em In Kicking And Screaming.

The Premise: Since we hadn't succeeded in getting teachers to open the doors to technological alternatives for their students, we'd have the students "come in through the back door" and throw oopen the doors for their teachers.

The Results: Quality and quantity of access to technology and the resulting products improved dramatically. Teachers began seeing the computer as a tool for the first time and capitalized on previously untapped capabilities (human and technological). The renaissance has touched every facet of the curriculum.

A Successful Model: State-of-the-Art Teaching Using an Integrated Learning System

Debora Doucette
Guam Department of Education

Abstract

Abstract unavailable at press time

Projects
World and Local Communities

Parent Night for National Geographic Kid's Network

Jill Greiner
Lancaster Country Day School, Pennsylvania

Abstract

On Wednesday, October 17, 1990 the parents of Lancaster Country Day School's fourth grade students learn from their children about the six week "HELLO" unit of the National Geogrpahic Kid's Network. The children have been investigating one aspect of our culture and behavior, pet ownership. The children have been student-scientists as they asked questions of themselves and other students from around the country and from other countries.

Using three computer rooms, the children become the teachers and instruct their parents in the various aspects of what they have learned. Some children teach their parents how to find the global address (longitude and latitude) of our school and of other schools on the team of student-scientist with which they have been co-operating. They use the computer and various globes and atlases. Other children show their parents how to use the computer to graph the results of their survey on pet ownership in our class. The parents have a chance to see firsthand some of the telecommunication messages we received and actually receive some messages themselves. This evening is a special project of communication with the parents. It allows them to experience firsthand what is done in the computer curriculum in our school. It is an opportunity for the children to be teachers and therefore to know if they really understand what we have been doing.

A Progress Report From the Marin School for Integrated Studies

Karen Randlev
Drake High School, California

Abstract

The mission of the Marin School for Integrated Studies is to help the student develop a sense of responsibility and productivity in a world that is increasingly complex and technological. To do this a redefinition of the school day, staff roles and student assignments is needed. Calls for these kinds of change are widespread. Through an initial partnership with Autodesk Inc. our high school has begun a model program for change. We are working to create a curriculum which integrates the teaching of science, English, and technology and enhances student experience through the devleopment of themes and community based projects. We will present examples of students' technology-based work; explain various integrated lessons and approaches; describe the community-based projects attempted. Evaluation of this first year of our experiment and future expansion will be shared.We are convinced that our concept for school reform will take both teachers and students in the right direction.

The Initial Charting of Computer Assisted Curriculum

Robert Hess
Arizona State University

Abstract

Abstract unavailable at press time.

Projects
How to Make Multimedia Come Alive

Multimedia in Maroa? Myth or Magic?

Kathy Bjelland
Maroa, Illinois

Abstract

Once upon a time, technology was a myth in Maroa, IL. A wonderful lady by the name of Karen, from the kingdom of CIESC, opened the eyes of two innocent teachers in this small humble school. Karen mesmerized these two hopefuls with the opportunities that were available. Through much hard work, powerful persuasion, and a few sleepless nights, the two hopefuls have transformed the once ordinary existence, into a magical, multi-medial enchantment for the young. Events of this tale span an era of four devoted, and sometimes frustrating, years.

Wizardry was used to hypnotize the reluctant. Kathy and Joyce used the spells of multi-media to convince wayward souls. The use of video equipment, calculators, overheads, and computers became a guiding light in the sometime dark and overwhelming tunnel of education. Realizing that the wide expanse of technology compared to the wee school of Maroa were often eons apart, the two innocent crusaders launched their attack. The villagers, the parents, and the young, joined the crusade.

The educational facility contains four computers, three VCRs, three televisions, a camcorder and limited software. These have been purchased with funds raised from local citizenry. Maroa Grade School proudly provides education for 235 children in Kindergarten through sixth grade. The spells cast by fully utilizing this equipment has shown the wonders of what small village schools do to provide for their young. An honest effort to prepare for the future has been met. There are many battles yet to be faced, more to be dreamed, and much to be accomplished. The desire and ability are present, now let us continue.

Happily ever after, everyone learned and learned.

Improving Instruction in Middle School Mathematics: Multimedia Teacher Training Project

Joyce Friske
Oklahoma State University

Abstract

Abstract unavailable at press time.

laking Your Curriculum Come Alive With Multimedia: Experiencing Multimedia's Magic!

ynn Lieberman
Oakland Schools,Waterford, MI

Abstract

This project focuses on the use of desktop video as a vehicle for students to compose, publish and present curricular topics. Thirteen teams of educators received intensive training in using and implementing a desktop video workstation. The challenge for the teachers was to train students to turn the technologies to their own use and to provide learning environments that actively engaged students in the learning process.

In the classroom projects that were initiated, students were encouraged to work collaboratively to research, plan and create curricular-focused video presentations that "published" their findings and communicated ideas and concepts. The project-oriented activities put collaboration, critical thinking, problem-solving, and oral and written communication skills into practice and allowed for creativity, imaginative spirit, and variety of presentation.

This presentation will include an overview of the teacher training component and will describe findings from the two-year project. Based on the experiences of the multimedia teams, considerations for establishing a multimedia environment, strategies for desktop video implementation in K-12 classrooms, and impact on teaching and learning will be shared. Examples of student-created presentations will be shown that demonstrate how multimedia can be used to enhance curriculum and student learning.

Projects
Logo Technology

Team Logo Meets Currie and Leitner

Camille Minichino, Stephen Sesko, Brian Webb
National Science Center, Lawrence Livermore National Laboratory, California

Abstract

Team Logo is a group of selected students, highly proficient in Logo programming, and motivated in learning. These boys and girls are unique in that they come from many different school districts to work together for three weeks during the summer, and the age range is very broad (grades four through eleven.) During the five years the group has been together, they have used Logo to investigate challenging and complex ideas in mathematics and science, including Fibonacci numbers, trigonometry, fractals, and chaos. They have worked on advanced programming ideas such as remodeling Logo commands, the use of global variables and recursion.

In 1989, Team Logo began to explore concepts of physics. They were introduced to the four basic forces of physics. They created a micro world in which to investigate the gravitational and the elctromagnetic fields. At NECC 90 we presented their work in modeling free fall, reflection, and refraction.

During the summer of 1990, the students continued their work in investigations of the weak force and the strong nuclear force. This year we will present a variety of simulations of radioactive decay, fission, and a program that demonstrates the technology of laser fusion.

This project in curriculum development was designed by the two presenters, one a computer education specialist, the other, a physicist. The project was implemented in a cooperative learning environment, where both the students and the mentors studied and learned together, sharing informationand solving problems.

The presenters will provide detail on the Team Logo project in general, and will demonstrate some of the student-authored programs from their study of physics.

Turtles and Toys: Technology in the Classroom

Deborah Bauder, Mary Planow, Ronald Sarner
State University of New York at Utica/Rome

Abstract

Project CHOICE (Curricular Help On Integrating Computers in Education) is a two year education and dissemination project funded through a $192,913 grant from the U.S. Department of Education's Office for Educational Research and Improvement under the Secretary's Fund for Innovation.This program is designed to serve as a model for the integration of computer technology across grade levels and subject areas, employing an easily learned yet sophisticated set of computer skills and tools. Teachers exposed to this training have been able to develop computer-based instructional activities in diverse subject areas. The learning environments created range from traditional computer-assisted instruction tutorials to discovery-based learning.

This presentation is an overview of the project with brief video highlights of classroom activities of some Project CHOICE participants. Handouts will include a more complete project description as well as a survey of lessons and materials developed by the participants.

What Personality Information Do Computer Facilitated Children's Drawings Reveal?

Judi Harris
University of Nebraska at Omaha

Abstract

To what extent can teachers discern verifiable personality information about young artists by looking at the children's creations? Do the media with which children construct their works affect the personally communicative properties of their pictures?

The scope and interjudge agreement of information communicated to computer-using classroom teachers through three types of children's drawings (freehand, graphics tablet, and Logo) were explored in a qualitative study. Logo-literate fifth-grade students were asked to draw three pictures each. The students, their teachers, and their parents were then interviewed to develop personality profiles. Profile content was compared to what Logo-using classroom teachers intuited about the children by looking at their artwork.

The study revealed several interesting result patterns:

- 69% of the perceptions that teachers stated about children after looking at their artwork were accurate.
- Drawing media type did not seem to affect the personally expressive content of children's drawings.
- Artist gender and age were not apparent to teachers viewing children's drawings.

During this project presentation, data samples and across-informant data summaries will be excerpted. Session participants will be given opportunities to view and respond to informants' pictures, then compare their intuitively received perceptions with artist personality profiles.

Projects
Apple II Applications

Bucks Facts, Inc.
Doug Lehnen
Bucks County, Pennsylvania

Abstract

Abstract unavailable at press time.

Apple IIs: The Most Ignored/Forgotten Hypermedia Tool
Richard Daughenbaugh
University of Southern Alabama

Abstract

Abstract unavailable at press time.

Technology for the Asking
John Hernandez
New York Institute for Special Education

Abstract

This presentation will discuss the process of setting up computer training seminars and the procedures to follow for getting loaner equipment from manufacturers at little or no cost. The presentation will consist of a multimedia presentation using the computer program called "Hyperstudio" on an Apple IIgs. The participants will get a sample letter of introduction, a checklist on dealing with manufacturers, and a sample student evaluation form.

The presentation will describe the following benefits of setting up training seminars:

1) extended loan perios for personal evaluation,
2) better purchasing decisions,
3) students get trained and evaluated befor e purchasing.

Projects
Higher Level Thinking Skills and Problem Solving

Computers for Higher Level Thinking and Cooperative Learning

Marlene Riley, M. Rogers
Tucson Unified School District, Arizona

Abstract

Abstract unavailable at press time.

Process Highlighters in a Computer Simulation: Facilitation of Theory-Oriented Problem Solving

Terrell N. Chandler and Christine Chaille´
University of Oregon

Abstract

This experimental study investigates whether or not process highlighters, incorporated in a computer simulation, facilitate the knowledge representation of early adolescent children. The two hypotheses addressed are (a) do process highlighters increase the percentage of subjects who employ theory-oriented problem solving when engaged in understanding a problem domain and (b) do process highlighters increase the percentage of subjects who demonstrate an understanding of the concept presented in the simulation?

Thirty-two 12 year old children from two middle schools played a computer game about light reflecting off flat mirrors, presenting the physical law that angle of incidence equals angle of reflection. Each child played one of two versions of the game for 40 minutes. One version incorporated process highlighters and one did not. During the session the students were encouraged to discuss and use diagrams to describe both their strategies for playing the game and their understanding of the concept embedded in the game. Each subject's transcripts were analyzed by three independent evaluators to determine their mode of explanation (random, concrete or hypothetical) and also their level of understanding (none, preliminary or good).

The results indicate that the use of process highlighters is positively correlated with the subject's level of understanding.

Investigating Problem Solving Strategies of Middle School Students Using Computer Software

Barbara Biglan
Pittsburgh, Pennsylvania

Abstract

The ability to solve problems effectively is given as a major goal at every level of educational endeavor. By studying the process of problem solution, teachers can recognize patterns of problem solving. Inadequate strategies can be recognized and the students can be taught more efficient heuristics. Successful skills can be reinforced and expanded.

Using two software programs, The Factory and the Incredible Laboratory from Sunburst Communications Inc., student's problem solving strategies were assessed. Those strategies which lead to solutions within a given number of steps were called successful. Strategies which lead to incorrect or partial solutions were labeled unsuccessful. The strategies were determined by examining worksheets which the students filled out while using the software. Solutions were analyzed in terms of the number of variables manipulated, the order of manipulation and the redundancy of variables. The differences between problem solving strategies and the successful solutions of boys and girls were analyzed.

Classroom teachers can understand a student's problem solving process by a relatively simple analysis of worksheets used in conjunction with certain software programs. This information provides the teacher with a basis for further instruction in the important process of successful problem solving.

Projects
Multimedia Faculty Development

Multi-Media Instructional Products

John H. Strange, Lora Kawka Davis, and Jennifer Smith

University of South Alabama

Abstract

For the past two years the College of Education has been attempting to stimulate the development of instructional projects and products using multi-media. We will use this presentation to demonstrate several of the multi-media projects and products that faculty and students have produced as a result of our efforts. Examples will be provided in the fields of Hypermedia, Art, Thinking Skills, Music, and Communications. An opportunity will be provided for participants to request samples and other information regarding our developmental efforts.

Transforming Instructional Strategies into Computer Courseware

Annette Lamb
University of Toledo, Ohio

Abstract

Abstract not available at press time.

The Multi-Media Cart: Getting Teachers To Use Technology in Teaching

John H. Strange
University of South Alabama

Abstract

Teachers, especially education college faculty, are quite willing to teach their students about technology, and to encourage their students to use technology in instruction when they become teachers in their own right. But few faculty actually use technology in their own classrooms.

We have developed at the College of Education of the University of South Alabama a Multimedia Cart, and an inservice program for faculty, that has succeeded in getting almost 40% of our instructional staff to actually use technology in their classrooms.

There are four basic components to the program: 1) strong leadership from the Dean; 2) our multimedia cart that puts the proper tools in the classroom easily; 3) a faculty development program beginning with a two hour session on making overheads on your computer and animating them for classroom use but can including 6 forty hour courses; and 4) extended support for faculty course development.

We will share with the audience the training activities that we have designed and the results we have had with the faculty; demonstrate a sample of the faculty computer instructional products (including some combining animation, interactive video and audio); and demonstrate our Multi-Media Cart."

Paper

CompuPASS: Computer-Based Assessment and Remediation of Academic and Test-Taking Skills

Vera T. Gershner
Texas Woman's University

C. Sue McCullough
Texas Woman's University

A Significant Problem

In 1990, three out of four jobs require a high school diploma *and* training *beyond* the high school level. Completion of post-secondary programs in two-thirds of the states is dependent upon successful performance on a standardized battery assessing minimum competency skills in reading, writing and mathematics. Four-fifths of both public and independent four-year colleges, and 87% of community colleges nationwide, report having minimum competency assessment initiatives underway (Fund for the Improvement of Post-Secondary Instruction, 1990). In Texas 28.3% of 20,461 students failed a mandated minimum competency test, the Texas Academic Skills Program (TASP) in September, 1990. Failure rates are disproportionate among minority and disadvantaged groups. TASP performance by ethnic group showed that 44.5% of Black students, 41.9% of Hispanic students, and 21.4% of Anglo students failed to pass at least one section of the TASP Texas Education Agency, 1991).

Huge costs for education and business are associated with remediating skill deficiencies. In Texas, 60% of adults living below the poverty level, 63% of adults on welfare, and more than 90% of the prison population are individuals who did not complete high school. Dropping out of school in Texas costs the state an estimated $17.12 billion in foregone income, lost tax revenue, unemployment insurance, adult training and education, and the increased cost of crime, welfare and incarceration (Texas Council for Vocational Education, 1988). Some means is needed to reduce costs while increasing the probability of success in reaching minimum levels of competency. Individualized computer systems of assessment and instruction offer one means to accomplish this goal.

Improving Existing Practice

It has been estimated that computer based instruction produces at least 30% more learning in 40% less time at 30% less cost compared to traditional classroom teaching (Perelman, 1990). Over 125 technologies and methods have produced at least double the productivity of traditional teaching methods—yielding at least twice as much learning for each unit of labor, cost, or time. These tools include the computer, VCR, television, audio tape players and even the telephone. The estimated cost to our nation of ignoring technology is estimated to be at least $100 billion per year (Perelman, 1990).

Public education has lagged behind corporate education which spends 300 times more on computer-based instruction than the public schools. Corporate employee education programs deliver most instruction with computers and telecommunications, achieving better results at less cost (Perelman, 1990). There are administrative barriers that impede change in public institutions. For example, public post-secondary schools base funding on average daily attendance (ADA) or number of students enrolled in formal classes, and therefore there is little incentive to improve learning efficiency in order to move students through the curriculum more quickly or to implement programs that allow self-paced learning outside the school walls or the formal class structure.

Further, faculty may resist changing traditional lecture-discussion teaching methods or traditional paper-pencil testing methods to incorporate and integrate technology and new teaching strategies into the curriculum since there are few rewards for the time, effort, risk and money expended to retrain themselves (Mecklenburger, 1990). Thus, students who had problems succeeding in traditional elementary and secondary schools, may find success just as elusive in traditional post-secondary schools.

Failure on Mandated Assessment of Academic Competencies

Texas law requires that school districts identify students at-risk for dropping out of school before graduation and provide programs designed to significantly reduce the dropout rate to 5% of their total student population by 1995. Criteria for identifying students at-risk for dropping out of school include: (a) failure on any one section of the Texas Assessment of

Academic Skills (TAAS), the mandated minimum competency test that must be passed at the 60% level to graduate from high school, (b) a failing grade in any one course, (c) retention in any grade, (d) referral for problem behavior and (e) high absenteeism. High schools have found that using these criteria, as many as 50-60% of their student body may be identified as at-risk for dropping out of school (Lilly, 1989). Since this law also denies driver's licenses to secondary school dropouts under the age of 18, the number of students seeking General Education Diploma (GED) courses is increasing. Passage of a standardized test of basic skills is required for a GED as well as for graduation from secondary and post-secondary public institutions demanding both basic academic and test-taking skills.

The Texas Academic Skills Program (TASP) is used at all public post-secondary institutions in TEXAS. While these institutions may *not* use the TASP to screen applicants for admission, students must take the TASP before accumulating nine (9) post-secondary level semester credit hours. Students *may not graduate* from a certificate program, an associate degree program, or a baccalaureate degree program until they have passed all three sections of the TASP: reading, writing and mathematics. For any skills not meeting the 70% competency level, the student must participate continuously in a remedial program in those skill areas and may *not* enroll in any upper-division courses (Texas Education Agency, 1989). Failure on the TASP, which measures skills at a twelfth grade level, has the potential to delay or end post-secondary education for at-risk students.

Students and instructors often do not know the *reasons* for failure on standardized examinations such as the TASP. Variables identified by researchers as related to failure include: *basic skills deficits* (Edwards & Waters, 1982; Fox, 1984, 1986), *lack of test-taking skills* (Bruch, Pearl & Giordano, 1986; Hughes & Douzenis, 1986), *learning disabilities* (Allard, Dodd & Peralez, 1987), *poor academic preparation* (Pedrini & Pedrini, 1986), *grade or course repetition* (Bianchi, 1984; Walker & Madhere, 1987), *lack of motivation* (Pascarella & Terenzini, 1983), and *reading comprehension difficulties* (Fox, 1984).

CompuPASS: Programmed Academic Skills and Strategies

Clearly, there is a need for an efficient and effective means to assess basic academic skills and effective test-taking skills, and to provide the necessary individualized instruction for students at-risk of failure. CompuPASS is one approach being developed and tested at the Technology and Learning Center (TLC) at Texas Woman's University. A telecommunication network, the CompuPASS Network, links TLC with selected secondary and post-secondary sites and enables educators across the state of Texas to develop and test CompuPASS as an effective assessment and remediation tool. An interdisciplinary Advisory Board provides guidance and support for CompuPASS.

Theoretical and Research Support

Nationwide students and teachers are increasingly involved with standardized tests despite their limitations on providing insights into teaching or learning. Grade point averages traditionally have been better predictors of post-secondary success than standardized tests (Pedrini & Pedrini, 1986). Educators need to know more about the process of problem solving involved in test taking rather than focusing only on the product of the assessment (Wagner & Sternberg, 1984). Cognitive and information processing theory suggests that the problem solving process is complex and is not accurately represented by a final product on a test (Sternberg, 1984). CompuPASS focuses on *basic skills deficits* and *test-taking skills* and provides a unique opportunity to study errors and decision making during testing. With the aid of computerized testing such data collection is possible (deMesquita, 1987; Ysseldyke, 1980). Every response of the individual is timed and recorded while the student takes the test on the computer.

Basic skills are *learned*, whether the skills are reading, writing, mathematics or test-taking skills (Rothman & Cohen, 1988; Samson, 1985; Scruggs & Marsing, 1988; Simpson, 1986). An error analysis of performance with CompuPASS in conjunction with information on specific student personal and demographic variables might be better predictors of student failure on the TASP and facilitate more effective planning for instruction and remediation strategies.

Description of Functions

Testing function . The testing function of CompuPASS assesses academic performance in reading, writing and mathematics and instantly provides both the student and teacher with data on strengths and weaknesses. In the Test Mode the student takes customized or computer-generated practice tests of self-selected length, in a self-selected area and then receives performance feedback immediately following the test. Performance feedback includes both content mastery and test-taking skill information. In the Study Mode the student gets immediate feedback on right and wrong answers as soon as each question is answered. The feedback includes detailed explanations explaining the correct answer. One research project underway is examining the effects of the four kinds of feedback on student performance.

CompuPASS not only provides the student with feedback on the correct response but also models logical thinking in the response explanations. In the Test Mode the student can choose to see the explanations after completing the test and reviewing the performance feedback. In the Study Mode, the explanations are available as soon as the student answers the question.

CompuPASS is being implemented in remedial or developmental classes where it is an adjunct to traditional instruction. CompuPASS may also be utilized in self-paced Personalized Systems of Instruction (PSI) which promote student responsibility for acquisition of concepts with minimal direct instruction. The PSI instructor serves in a facilitative role, providing guidance and support and direct instruction based only upon identified student needs.

CompuPASS's individualized, self-paced format may be less threatening to students who have been unable to compete effectively in traditional instructional environments, and to less competitive or non-traditional students. For example, the individualized, self-pacing may be more supportive of the learning style of Hispanic students whose traditional culture does not reinforce competitiveness and attention to time variables. Non-traditional students, such as older women, may be less threatened by a more supportive self-paced environment as well.

CompuPASS utilizes an interactive format that requires the user to actively participate in learning, in contrast to the passive learning that occurs with book-based instruction. This interactive format promotes attention and motivation to stay on task. A large database of questions provides multiple trials without repetition of content and provides exposure to varying content and difficulty levels. For example, reading levels in the reading and writing sections vary from eighth to sixteenth grade equivalent levels. Each question is coded as to reading level and is analyzed for item difficulty and item relationship to prediction of the final score.

CompuPASS provides diagnostic analysis of deficiencies in reading, writing and mathematics. The content mastery report includes the percentage and number correct for each general area, as well as for specific sub-skills in each area; six subskills in reading, five in writing and ten in mathematics.

Test-taking skills assessment function . A significant characteristic of CompuPASS is its ability to trace a student's actions as the individual takes the test, allowing examination of the pattern of the student's test-taking decisions and providing insight into learning and test-taking processes that is not possible with paper and pencil tests. For example, CompuPASS identifies variables that were related to the production of incorrect answers. Two such variables are pacing and changing answers. CompuPASS reports the amount of time spent on correct and incorrect responses in each area, the total amount of time in each area, and the number of changed answers in each area that were changed from right to wrong, wrong to right, or wrong to wrong.

Students with marginal academic skills may be doubly jeopardized by poor test-taking skills or a lack of experience with standardized tests. CompuPASS provides feedback on test-taking skills as well as repeated practice with a simulated standardized test. For example, CompuPASS informs a student s/he has changed right answers to wrong in reading and perhaps the reverse in math. Informed students can change their behavior and observe the results immediately.

The CompuPASS sequential record charts the student's actions. Guessing or test anxiety patterns may appear as multiple changed responses. Impulsive responses may be indicated by short answer times on incorrect answers. Answers that are considered for a longer period of time than average probably are not as well mastered as those that are answered quickly and correctly. Pacing strategies are evident in the amount of time given to each item, and the number of changes to previously considered items on the test. Illogical thinking may be deduced from studying the patterns of incorrect choices (Wagner & Sternberg , 1984). "Analyses of response times and patterns of errors in a wide variety of problems can provide the keys to unlock at least some of the mysteries of the mind" (Sternberg, 1986). CompuPASS provides instant individual and group feedback on test-taking actions that may assist in planning effective remediation and instructional strategies.

CompuPASS is being used in adult and compensatory education to provide feedback to individuals on academic strengths and weaknesses, and to provide practice and instruction in taking standardized multiple choice tests. CompuPASS may be used to desensitize or reduce high test anxiety through repeated practice with positive feedback in an environment that closely simulates standardized testing conditions (Tobias, 1985). It has the potential to be a powerful diagnostic tool for students with learning disabilities (Scruggs & Lifson, 1986; Scruggs and Mastropieri, 1986).

Evaluating CompuPASS

Educators from across the state of Texas developed the minimum reading, writing and mathematics competencies tested on the TASP reflecting national research on minimum competencies required for success in higher education. Evaluation is tied directly to assessing CompuPASS as (a) a predictor of successful TASP performance, (b) a reliable and valid individual assessment procedure, (c) an effective remediation technique for at-risk students and (d) a technically superior product. These evaluations lead directly to increasing the sophistication and contributions of CompuPASS. The program is being evaluated as a supplement to developmental education classes in eight sites across Texas.

Since CompuPASS produces information not available on paper-pencil assessments, one research project currently underway is analyzing test-taking data statistically to determine if clusters or patterns of behavior exist, and if so, what the relationship of these patterns is to performance on CompuPass and the TASP.

Formative evaluation is occurring throughout the research. Extensive objective data is being collected to validate the TASP program. Evaluative questionnaires are collected from every participant, student and instructor, on the software, hardware and procedures followed. Software and procedures will be modified based on the feedback received. Evaluation criteria include technical factors such as ease of use, error checking, interface design, and program design; content factors such as clarity and usefulness of feedback, quality and nature of questions and feedback; and motivational or interest factors such as screen presentation aesthetics, engaged time on task and interest level.

Network members will be asked to evaluate TASP test and lesson items within their area of expertise for quality; relationship to stated objectives; feedback accuracy; gender, age and racial bias; fairness; variety; reading level and content validity. Every passage is checked with *Right Writer* (1990) which measures reading level, and checks spelling and grammar. Correlations of CompuPASS with TASP performance will be used to statistically analyze the concurrent and content validity of CompuPASS.

CompuPASS, as a remediation and preparation aide, is being held to high standards of validity and reliability but obviously will not have the legal impact that TASP must have. CompuPASS will be used by secondary schools for preparation for the TASP, as well for remediation, and therefore it must show strong correlations with the TASP examination.

CompuPASS is being evaluated as a remediation strategy when it is used as an adjunct to traditional instruction. Planned research will also evaluate using CompuPASS within a Personalized System of Instruction (PSI), a less structured but more individualized learning environment. This information will be important for assessing its effectiveness as a remediation strategy for students who have failed the TASP.

These scientific evaluations will provide important information with which to judge the cost effectiveness, efficiency, acceptability and pedagogical effectiveness of CompuPASS as well as to refine the technical aspects and functions of the program. CompuPASS will be demonstrated and research results discussed in the presentation at the National Educational Computing Conference.

References

Allard, W. G., Dodd, J. M., & Peralez, E. (1987). Keeping LD students in college. *American Therapy*, 22(4), 359-365.

Bianchi, S. M. (1984). Children's progress through school: A research note. *Sociology of Education*, *57*, 184-192.

Bruch, M. A., Pearl, L. & Giordano, S. (1986). Differences in the cognitive processes of academically successful and unsuccessful test-anxious students. *Journal of Counseling Psychology*. *33*(2), 217-219.

deMesquita, P. (1987, March). *The information processing and diagnostic decision making of school psychologists while solving a computer-simulated diagnostic referral problem.* Paper presented at the National Association of School Psychologists Convention, New Orleans, LA.

Edwards, J. E., & Waters, L. K. (1982). Involvement, ability, performance, and satisfaction as predictors of college attrition. *Educational and Psychological Measurement*, *42*(4), 1149-1152.

Fox, R. N. (1984). Reliability and discriminant validity of institutional integration scales for disadvantaged college students. *Educational and Psychological Measurement*, *44*(4), 1051-1-57.

Fund for the Improvement of Post-Secondary Instruction, (1990) *Comprehensive Program information and Application Procedures Fiscal Year 1991*. U.S. Office of Post-Secondary Education.

Hughes, T. M., & Douzenis, C. (1986). Predictor and performance variables in a performance-based education course. *Journal of Psychology*, *120*(2), 143-147.

Lilly, B. (1989). A Comparison of Counseling Interventions with At-Risk High School Students. Doctoral Dissertation (in progress). Texas Woman's University. Dr. Karen Jackson, Dissertation Chairperson.

Mecklenburger, J.A. (1990). Educational Technology is not enough. Phi Delta Kappan, 72(2), 105-108.

Pascarella, E. T., & Terenzini, P. T. (1983). Predicting voluntary freshman year persistence/withdrawal behavior in a residential university: A path analytic validation of Tinto's model. *Journal of Educational Psychology*, *75*(2), 215-226.

Pedrini, B. C., & Pedrini, D. T. (1986). Attrition/persistence: Stepwise multiple predictors for disadvantaged and control freshmen. *Journal of Instructional Psychology*, *13*(3), 131-134.

Perelman, L. J. (1990). *Luddite schools wage a wasteful war.* The Wall Street Journal, Monday, September 10.

RightWriter Macintosh Version 3.1 (1990). RightSoft Incorporated, 4545 Samuel, Sarasota, Florida, 34233 [software]

Rothman, E. Q. & Cohen, J. (1988). Teaching test taking skills. *Academic Therapy*, *23*(4), 341-348.

Samson, G. E. (1985). Effects of training in test-taking skills on achievement test performance: A quantitative synthesis. *Journal of Educational Research*, *78*(5),261-266.

Scruggs, T. E. & Lifson, S. (1986). Are learning disabled students "test-wise?" An inquiry into reading comprehension test items. *Educational and Psychological Measurement*, *46*(4), 1075-1082.

Scruggs, T. E. & Marsing, L. (1988). Teaching test-taking skills to behaviorally disordered students. *Behavioral Disorders*, *13*(4), 240-244.

Scruggs, T. E. & Mastropieri, M.A. (1986). Improving the test-taking skills of behaviorally disordered and learning disabled children. *Exceptional Children*, *53*(1), 63-68.

Simpson, M. L. (1986). Teaching university freshmen to employ, regulate and transfer study strategies to the content areas. *Forum for Reading*, *17*(2),61-71.

Sternberg, R. J. (1984). Toward a triarchic theory of human intelligence. *Behavior and Brain Sciences*. *7(2)*, 269-315.

Sternberg, R. J. (1986). Inside intelligence. *American Scientist*, *74*(2), 137-143.

Texas Council for Vocational Education, 1988 *The Dropout Dilemma Searching for Formulas that Work.* Position paper. Austin, TX: Author

Texas Education Agency, (1991). *Texas Academic Skills Program test results from September 22, 1990.* TASP™: FYI Newsletter of the Texas Academic Skills Council, Number 9, Spring.

Texas Education Agency (1989). *1989 TASP Test Registration Bulletin*. Austin, TX: Texas Higher Education Coordinating Board, Texas Education Agency, National Evaluation Systems.

Tobias, S. (1985). Test anxiety: Interference, defective skills, and cognitive capacity. *Educational Psychologist*, 20(3), 135-142.

Walker & Madhere, (1987). Multiple retentions: Some consequences for the cognitive and affective maturation of minority students. *Urban Education*. 22(1), 85-102.

Wagner, R. K., & Sternberg, R. J. (1984). Alternative conceptions of intelligence and their implications for education. *Review of Educational Research*, *54*(2), 179-223.

Ysseldyke, J. E. (1980). Technical adequacy of tests used by professionals in simulated decision-making. *Psychology in the Schools*. *17*, 202-209.

Paper

Teaching Cognitive Monitoring Strategies Through Guided Logo Instruction

Mi Ok C. Lee
Ann D. Thompson
Iowa State University

Abstract

Research supports that explicitly designed Logo instruction stimulating students' thinking processes may lead students to the development of cognitive monitoring strategies. The guided Logo instruction developed in this paper provides an instructional model for using Logo programming as a vehicle to teach cognitive monitoring strategies. In this guided Logo instruction, three pedagogical elements are linked together: an instructional plan which explicitly models cognitive monitoring strategies, the use of Logo programming as a tool for providing a dynamic and challenging learning environment, and teacher mediated activities applying such strategies to general problem situations. These pedagogical elements are all addressed in the Logo instruction. Thus, this guided Logo instruction helps students activate their cognitive system and deliberately monitor on-going thinking processes. The model of cognitive monitoring activities applied to solve a Logo problem consists of five steps: decomposing a given problem, planning a solution, executing the plan, identifying errors and explaining causes of errors, and debugging errors on a solution. The teacher cooperative activities encourage students to consciously engage in the formal cognitive monitoring process and to cycle the process recursively until they solve the problem. Teacher mediated practice of cognitive monitoring strategies in everyday problem situations, which is modeled in this instruction, helps students transfer the strategies learned in Logo programming to other situations.

Introduction

The expansion of knowledge and technology demands that schools dedicate more attention to the creation of educational environments that can help students develop thinking skills (Maddux & Johnson, 1988; Task force on Teaching As a Profession, 1986; Swan & Black, 1989). Cognitive monitoring is one of the critical factors of individual differences in intelligent and efficient thinking (Brown, 1983; Sternberg, 1984; 1987). Further, cognitive monitoring has been regarded as one of the most important and essential strategies for efficient problem solving (Brown, 1978; Lochhead, 1989; Ford, 1981). Since cognitive monitoring strategies are considered essential for efficient thinking and problem solving, it is important to develop proper instructional methods and learning tools to help students develop general cognitive monitoring strategies.

Teaching and learning computer programming can provide a dynamic and challenging learning environment. It has been argued that this environment can help students improve a broad range of problem solving abilities. In particular, it has been suggested that Logo programming can be an excellent means for developing problem solving strategies and thinking skills (Papert, 1980; Watt, 1982).

Well-structured instruction with teacher mediated learning helps students manage the complex information to be learned. Teacher mediated learning also provides an opportunity for students to mindfully engage in the acquisition of cognitive skills and metacognitive skills and to transfer those learned skills to other domains of learning (Littlefield, Delclos, Lever, Clayton, Bransford, & Franks, 1988; Salomon & Perkins, 1987; Singley & Anderson, 1989; Swan & Black, 1989).

The guided Logo instruction developed in this paper provides a model of using Logo programming as a vehicle to teach cognitive monitoring strategies. In this Logo instruction, three main pedagogical elements are linked together: 1) an instructional plan which explicitly models cognitive monitoring strategies; 2) the use of Logo programming as a tool for providing a dynamic and challenging learning environment which is necessary in order for students to consciously monitor their ongoing thinking processes; and 3) teacher mediated practice applying such cognitive monitoring strategies to general problem situations. The pedagogical elements in this instructional design target the development of general cognitive monitoring as well as learning Logo programming. The teacher mediated practice facilitates the transfer of cognitive monitoring strategies developed in Logo programming to other learning domains.

Rationale

Cognitive monitoring is typically defined as the ability to regulate or evaluate one's cognitive process and to direct ongoing thinking processes (Flavell, 1981; Forrest-Pressley and Waller, 1984; Brown, 1978; Baker and Brown, 1987). Everyone uses cognitive monitoring to deal with everyday problems. However, good and poor problem solvers differ markedly in selecting strategies for problem solving. As compared to poor problem solvers, good problem solvers more often use explicit cognitive monitoring strategies such as defining a problem, predicting an outcome, planning ahead, regulating the process, checking errors, and modifying their thinking processes (Derry, 1989; Wagner & Sternberg, 1987). It has been suggested that today's college students and young American adults lack the cognitive monitoring strategies necessary to become independent problem solvers and efficient thinkers. Many college students have great difficulty simply managing and evaluating their own learning efforts (Chipman & Segal, 1985; Baker, 1989; Shoenfeld, 1985).

Although there is a great need to improve students' use of cognitive monitoring strategies, there has been little research on effective instructional methods and learning tools that can facilitate the development of students' cognitive monitoring strategies.

General cognitive monitoring training is difficult to incorporate into the classroom; most school curricula and materials have not been developed to elicit cognitive monitoring activities while students are engaged in learning domain specific content. Although teachers and students may often interact with each other by questioning and answering, such interactions are rarely seen by students as incorporating general and systematic cognitive monitoring skills.Classroom teaching which focuses on improving cognitive monitoring strategies needs to make such a process more explicit and apparent to students. Instruction may need to be formally structured to help students become aware of the general cognitive monitoring processes. Formal instruction of cognitive monitoring strategies may assist students in effectively utilizing and transferring them to appropriate situations.

Computer programming is a potential tool with which students can naturally manipulate an environment and develop cognitive skills and metacognitive skills. In particular, Logo has been cited as a powerful vehicle for teaching students about their own thinking processes (Papert, 1980). The Logo environment allows students to engage in dynamic activities, such as selecting their own goals, outlining a solution, and testing hypotheses. Also, immediate and clear error messages in Logo motivate students to debug their errors (Clements, 1987, 1990). These characteristics of Logo give it the potential of being a powerful tool to activate a student's cognitive monitoring strategies. However, there is conflicting evidence of the claims for developing problem solving strategies through programming (e.g. Pea & Kurland, 1984; Salomon & Perkins, 1987). Recently Maddux and Johnson (1988), Seidman (1987), and Swan and Black (1989) have claimed that teaching Logo programming alone is not sufficient to guide students toward the development of higher level cognitive skills. Teachers can not improve metacognitive strategies by simply teaching students to write Logo programs. Logo programming combined with guided instruction is needed to help students acquire more than just technical programming skills (Leron, 1985; Seidman, 1987).

In general, researchers agree that for the positive transfer of problem solving skills to be achieved, well defined, specific problem solving strategies and explicitly structured instructional treatment should be incorporated into the mediated learning environment of Logo programming (Leron, 1985; Pea & Kurland, 1984; Salomon & Perkins, 1987; Swan & Black, 1989). Yet, regardless of the claim for the importance of the instructional method in Logo, there is little existing Logo instruction which provides a concrete model for teaching cognitive monitoring strategies. In order to maximize the potential of Logo programming experiences, guided Logo programming instruction needs to be developed to facilitate the formal acquisition of cognitive monitoring strategies and to provide the transfer for such strategies to other domains.

Overview of Approach to Develop Logo Instruction

This paper presents Logo-based instruction developed to teach cognitive monitoring strategies to pre-service teachers at Iowa State University. In this instructional model, an explicit model of cognitive monitoring activities is taught within the context of teaching Logo programming. The model allows students to naturally activate their cognitive monitoring of ongoing thinking processes. The model of cognitive monitoring consists of five steps: decomposing a given problem, planning a solution, executing the plan, identifying errors and explaining causes of errors, and debugging errors. The instructional process focuses on students' interaction with the teacher by providing Socratic dialogues in the process of solving Logo problems. The teacher uses a Socratic dialogue technique to alert students to follow formal cognitive monitoring activities recursively. Teacher mediated practice for solving everyday problems is also modeled in this instruction in order to encourage the transfer of learned cognitive monitoring strategies to other domains.

Target audience. The target audience is 140 students enrolled in Secondary Education 101, at Iowa State University in Fall, 1990. This class, titled "Educational Applications of Computers", is a 3 credit course for pre-service teachers. This educational computing class is designed to help students develop basic skills for using computers in instruction. Although this instruction has been tested with college students, the general approach to teach Logo programming in conjunction with the general nature of cognitive monitoring can be adopted to elementary and high school students.

Instructional Goals. This instruction is designed to help students learn the Logo programming language and develop general cognitive monitoring strategies.

Acquisition of Logo programming skills can be accomplished by:

1. Understanding the philosophy of Logo programming, the structure of Logo language, and the features of a procedural programming language;

2. Understanding Logo primitives and utilizing them;
3. Understanding the rationale of learning procedures, the concept of super procedure and subprocedure, and being able to write procedures in Logo;
4. Understanding the concept of the total turtle trip theorem and being able to utilize the theorem in Logo;
5. Understanding the rationale of learning variables, the concept of variable, and being able to use variables in Logo;
6. Understanding the rationale of learning recursion, the concept of recursion, and being able to use recurision in Logo;

The development of general cognitive monitoring strategies can be accomplished by facilitating students' improvement of:

1. Decomposing skills which are defined as breaking a complex graphic into manageable units of simpler graphics;
2. Planning skills which are defined as the ability to organize and sequence each component of decomposed problems in order to devise an efficient solution;
3. Executing skills which are defined as carrying out a planned solution;
4. Identifying problem skills which are defined as the ability to identify discrepancies between a planned graphic and the actual outcome, to indicate error procedures, and to explain the problem with the error procedures;
5. Debugging skills which are defined as the ability to correct errors. (figure 1)

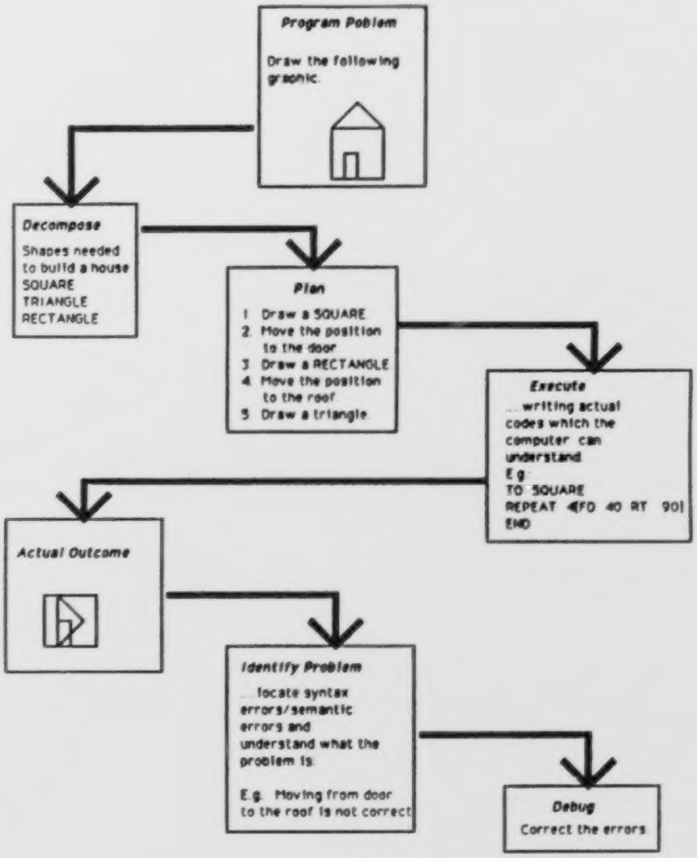
Organization of the Instruction (see Appendix A)

1. Instructional content unit

Based on the theoretical framework presented above, a unit designed to teach Logo programming using a guided instructional approach that incorporates associated hands-on activities has been constructed. The unit includes the teaching of the five cognitive monitoring strategies; decomposing, planning, executing, identifying problem, and debugging (Figure 1). The content for Logo programming has been organized into five main units: primitives, procedures, total turtle trip theorem, variables, and recursion.

A. Unit One: Introduction to Logo programming and Cognitive Monitoring

This unit is designed to introduce Logo programming and cognitive monitoring. The unit presents students with reasons


Program Poblem
Draw the following graphic.
Decompose
Shapes needed to build a house
SQUARE
TRIANGLE
RECTANGLE
Plan
1 Draw a SQUARE
2 Move the position to the door
3 Draw a RECTANGLE
4 Move the position to the roof
5 Draw a triangle
Execute
...writing actual codes which the computer can understand
E.g.
Actual Outcome
Identify Problem
...locate syntax errors/semantic errors and understand what the problem is.
E.g. Moving from door to the roof is not correct
Debug
Correct the errors

```
TO SQUARE
REPEAT 4[FD 40 RT 90]
END
```

Figure 1. Cognitive Monitoring in Logo Programming

BEST COPY AVAILABLE

1. Decompose	Break a complex problem into smaller, self-contained problems and solve each part correctly
2. Plan	Consciously organize and sequence the decomposed problems to devise the solution to the problem efficiently
3. Execute	Test planned hypotheses or solutions.
4. Identify Problem	Compare the actual outcome with the planned graphic, locate errors in the procedures, and explain the causes of detected errors.
5. Debug	Correct the detected errors or misconceptions.

Figure 2. 5 Basic Components of Cognitive Monitoring

for learning Logo programming and introduces the importance of cognitive monitoring strategies for solving a novel problem efficiently. The unit consists of three main areas. First, a philosophy of learning Logo programming and development of thinking skills through Logo programming is introduced. Second, a definition of cognitive monitoring is presented and applied to an everyday problem; then, the benefits of learning cognitive monitoring strategies are presented. The everyday problem provides students with a general understanding of how they can be involved in cognitive monitoring activities in their life. Third, the five basic component processes of cognitive monitoring are introduced (Figure 2). Then, an example of an everyday problem is presented to which the five cognitive monitoring processes are applied in order to solve the problem (Appendix B). At the end of the unit, Logo primitives are introduced.

B. Unit Two: Introduction to Procedures

This unit is designed to teach the concepts of super and sub procedures and to teach students to write procedures.

First, students are taught to divide complex information into manageable sub-parts. Most problems which students deal with are larger and more complex than can be effectively handled. Decomposition of complex problems into manageable "mind-size" bites is extremely important for successful problem solving. The idea of learning super-subprocedures helps students organize and sequence a set of decomposed information in order to devise a meaningful product and effectively handle a complex problem.

The content unit consists of five parts: rationale for learning procedures; demonstration of writing super procedures and subprocedures; a teacher-student cooperative Logo activity; a student activity; and the presentation of an everyday problem with which to practice cognitive monitoring activities.

C. Unit Three: Introduction to the total turtle trip theorem

This unit is designed to introduce the total turtle trip theorem. The unit is designed to teach students to understand angles and definitions of polygons in a meaningful and relevant way. It is also intended to help students improve their spatial abilities, such as orientation, rotation, and direction.

The unit consists of five parts: rationale for learning the total turtle trip theorem; demonstration of the theorem by drawing various regular polygons; a teacher-student cooperative Logo activity; a student activity; and the presentation of an everyday problem to practice cognitive monitoring activities.

D. Unit Four: Introduction to Variables

This unit is designed to introduce the concept of using variables in Logo programming.

The main goal of this unit is to improve a student's ability to look for a general solution to a specific problem. Teaching variables in Logo programming helps students apply a certain solution to new and different problem situations. The unit is also designed to teach students to understand the importance of abstraction and generalization in mathematics.

The content unit consists of five parts: rationale for learning variables; demonstration of variables in Logo programming; a teacher-student cooperative activity; a student activity; and an example of everyday problem to apply the learned cognitive monitoring strategies to general problems.

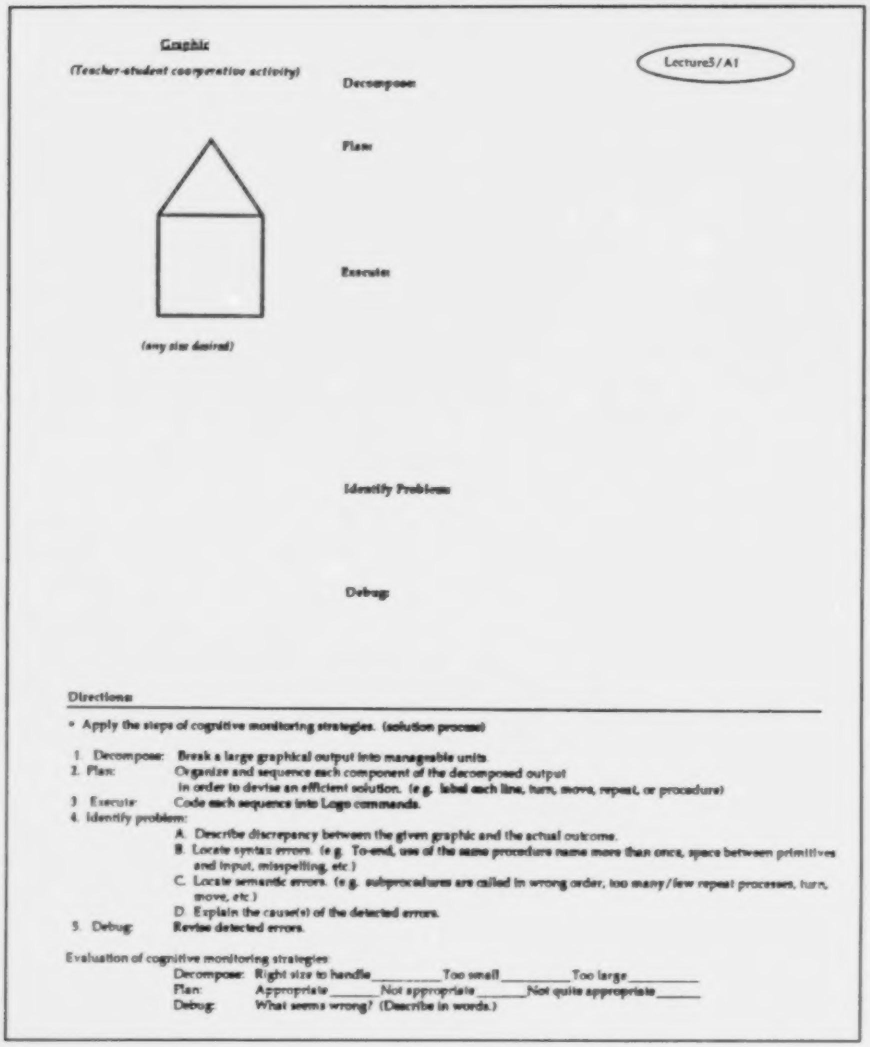
E. Unit Five: Introduction to Recursion

This unit is designed to teach recursion in Logo programming. Recursion is a powerful concept in computer science. Teaching recursion helps students write more efficient and compact programs in order to produce relatively complex graphics.

The unit consists of five parts: rationale for learning recursion; introduction to recursion in Logo programming; a teacher-student cooperative activity; a student activity; and an example of everyday problem to utilize components of cognitive monitoring in everyday life.

2. Instructional setting

The instruction was first used in the Fall, 1990, starting in September and continuing into October. The instruction was also used during Spring, 1991. The instructional duration is four weeks, a total of 16 hours. Each week, students receive two hours of lecture and two hours of laboratory instruction. In addition, the students receive a minimum of two hours of homework each week. The lecture provides broad concepts and



Graphic

(Teacher-student cooperative activity)

Lecture3/A1

(any size desired)

Decompose:

Plan:

Execute:

Identify Problem:

Debug:

Directions:

- Apply the steps of cognitive monitoring strategies. (solution process)

1. Decompose: Break a large graphical output into manageable units.
2. Plan: Organize and sequence each component of the decomposed output in order to devise an efficient solution. (e.g. label each line, turn, move, repeat, or procedure)
3. Execute: Code each sequence into Logo commands.
4. Identify problem:
   - A. Describe discrepancy between the given graphic and the actual outcome.
   - B. Locate syntax errors. (e.g. To-end, use of the same procedure name more than once, space between primitives and input, misspelling, etc.)
   - C. Locate semantic errors. (e.g. subprocedures are called in wrong order, too many/few repeat processes, turn, move, etc.)
   - D. Explain the cause(s) of the detected errors.
5. Debug: Revise detected errors.

Evaluation of cognitive monitoring strategies:

Decompose: Right size to handle__________Too small__________Too large__________

Plan: Appropriate_______Not appropriate_______Not quite appropriate_______

Debug: What seems wrong? (Describe in words.)

Figure 3. Cognitive Montoring Sheet

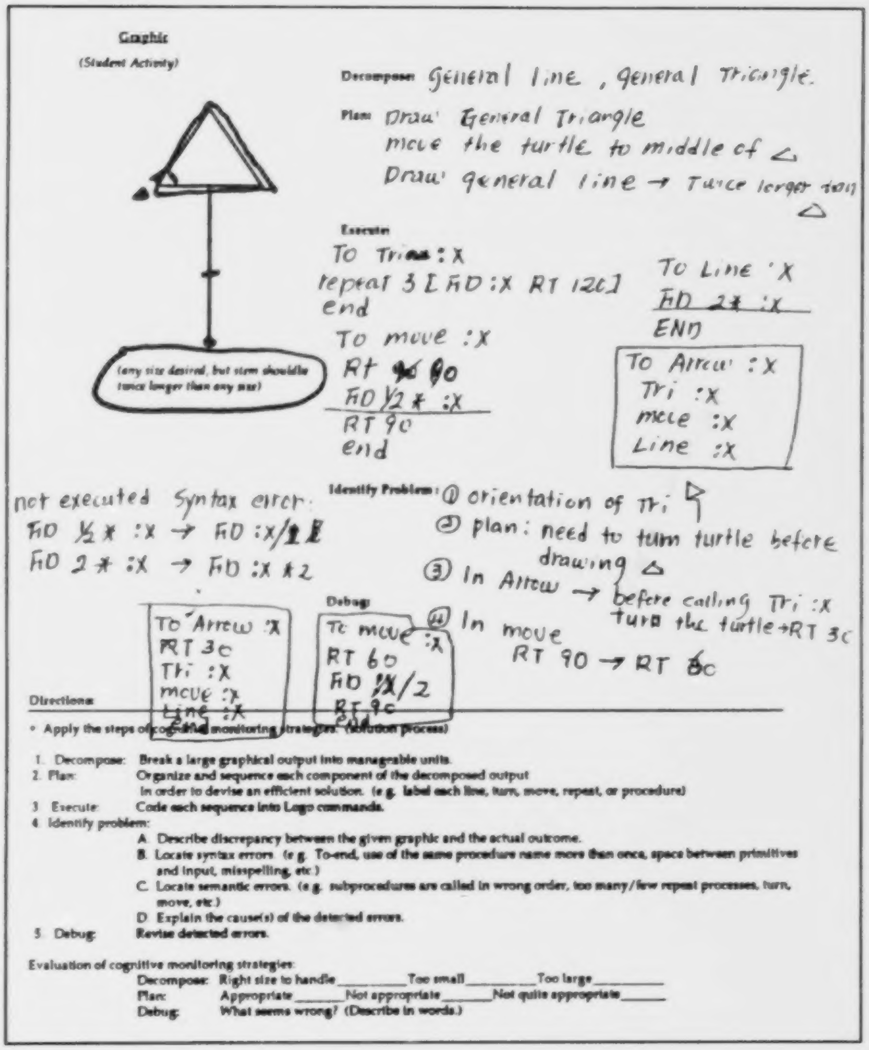
rationale, and group activities for solving Logo problems with the utilization of cognitive monitoring activities. In the laboratory, students work on the computer individually to solve given Logo problems using cognitive monitoring. Each activity is designed to allow students to apply the five component processes of cognitive monitoring.

3. Instructional procedure

The students receive guided Logo programming instruction. In this guided instruction, students are presented with a step by step approach to solving a given programming problem. First, students work with an instructor cooperatively to utilize formal cognitive monitoring activities (Figure 3). The teacher intentionally raises a Socratic dialogue to alert students to deliberately engage in cognitive monitoring activities. Then, the students alone utilize formal cognitive monitoring activities through working on student activity worksheets (Figure 4). Finally, the instructor provides problems from various subjects and encourages students to transfer formal cognitive monitoring strategies learned in Logo programming to the general problem situations (Appendix C).

This guided Logo instruction emphasizes the acquisition of metacognitive skills in the process of solving programming problems and transferring these metacognitive skills to other domains. Specifically, the instruction focuses on 1) the development of general cognitive monitoring strategies as one particular problem solving process, 2) an explicit instructional model activating cognitive monitoring strategies, and 3) teacher-student cooperative learning as one type of mediated learning technique.

BEST COPY AVAILABLE



Graphic

(Student Activity)

(any size desired, but stem shouldbe twice longer than any size)

Decompose: general line, general Triangle.

Plan: Draw General Triangle
move the turtle to middle of △
Draw general line → Twice larger than △

Execute:

```
To Tri :X
repeat 3 [ FD :X RT 120]
end
To move :X
Rt 90
FD 1/2 * :X
RT 90
end
```

```
To Line :X
FD 2* :X
END
```

```
To Arrow :X
Tri :X
move :X
Line :X
```

not executed Syntax error:
FD 1/2 * :X → FD :X/2
FD 2 * :X → FD :X * 2

Identify Problem: ① orientation of Tri
② plan: need to turn turtle before drawing △
③ In Arrow → before calling Tri :X turn the turtle → RT 30
④ In move
RT 90 → RT 60

Debug:

```
To Arrow :X
RT 30
Tri :X
move :X
Line :X
end
```

```
To move :X
RT 60
FD :X/2
RT 90
end
```

Directions:

- Apply the steps of cognitive monitoring strategies. (solution process)

1. Decompose: Break a large graphical output into manageable units.
2. Plan: Organize and sequence each component of the decomposed output in order to devise an efficient solution. (e.g. label each line, turn, move, repeat, or procedure)
3. Execute: Code each sequence into Logo commands.
4. Identify problem:
   - A. Describe discrepancy between the given graphic and the actual outcome.
   - B. Locate syntax errors (e.g. To-end, use of the same procedure name more than once, space between primitives and input, misspelling, etc.)
   - C. Locate semantic errors. (e.g. subprocedures are called in wrong order, too many/few repeat processes, turn, move, etc.)
   - D. Explain the cause(s) of the detected errors.
5. Debug: Revise detected errors.

Evaluation of cognitive monitoring strategies:
Decompose: Right size to handle______ Too small______ Too large______
Plan: Appropriate______ Not appropriate______ Not quite appropriate______
Debug: What seems wrong? (Describe in words.)

Figure 4. Cognitive Montoring Sheet

1) *Development of general cognitive monitoring as one particular problem solving process:* Each lesson outline emphasizes the cognitive monitoring process inherent in Logo programming, rather than the programming activity itself. Students are encouraged not to jump into programming on the computer, but rather to sequentially follow the steps of the cognitive monitoring process.

Two in-class programming problems are presented during each lecture. The first problem allows students to work cooperatively with the instructor. The instructor models appropriate cognitive monitoring processes. This activity takes about 15 minutes. Then, the students alone work the second Logo programming problem by engaging in the formal cognitive monitoring activities. The instructor, by active Socratic questioning and observing, encourages students to consciously follow step-by-step these cognitive monitoring processes as outlined on the instructional worksheets (Appendix D).

The instructor emphasizes that using Logo programming activities provides a way to practice the general cognitive monitoring process. It is also emphasized that learning this general cognitive monitoring process is an important goal for computer programming instruction. Thus, students are encouraged to utilize cognitive monitoring in computer programming.

2) *Explicit instructional model activating cognitive monitoring strategies:* Formal cognitive monitoring activities are structured in order to provide direct and explicit instruction to teach cognitive monitoring strategies. The five components of the formal cognitive monitoring process are de-

composing, planning, executing, identifying the problem, and debugging.

The decomposing activity is designed to help students search for definite and necessary elements or segments of a complex graphic problem. If there are no absolute segments of graphics or elements, students need to find critical clues that make the graphic less complex. Developing decomposing strategies is a fundamental skill students need to acquire to become good problem solvers.

The planning activity is designed to help students organize and sequence the decomposed information and elements in order to solve the given problem. Most important, students need to search for strategies to develop an efficient solution to the problem. They also need to consider structure and modularity of the program. Thus, the planning activity helps students understand important concepts of repeat, procedures, total turtle trip theorem, variables, and recursion and apply those concepts to develop an efficient solution to the programming problem. Through the planning process, students deliberately monitor their plan.

The executing activity is designed to help students translate the planned solution into Logo programming commands and execute them on the computer. The Logo programming environment allows students to carry out the result immediately.

The identifying problem activity is designed to encourage students to describe the discrepancies, if any, between the desired graphic and the actual outcome. This is an important first step to identifying errors, as it prevents students from searching for errors mindlessly by doing trial and error corrections without understanding the causes of the errors. Describing the discrepancies also helps students more easily locate the statements which might cause the errors. For example, if the discrepancy is the orientation of the graphic, then students would look at the turtle's turn, rather than the distance. As a second step of identifying the problem, students need to locate procedures that might have caused critical errors. Finally, students need to state what is wrong with the procedures they located. Through the identifying problem activity, students consciously attempt to understand their misconception or thinking errors.

In the debugging activity, students are allowed to correct errors on the computer.

Following these steps, students pursue cognitive monitoring activities recursively until they solve the given problem. Through the recursive cognitive monitoring activities, students reformat their original strategy or method and refine the solution (Figure 4 and Figure 5).

3) Mediated learning: In this guided Logo instruction, the instructor intentionally raises Socratic dialogues to help students concentrate on formal cognitive monitoring activities in the process of solving Logo problems (Appendix E). With the instructor's guidance, students work through five steps of formal cognitive monitoring activities on worksheets which are designed to elicit those cognitive monitoring strategies.The teacher collects common decompositions of a given Logo problem by asking Socratic questions. In the planning step, students are encouraged to search for an efficient solution process. Then, they are allowed to translate the plan to Logo commands. Much time is spent in the decomposing and planning step rather than the executing step.

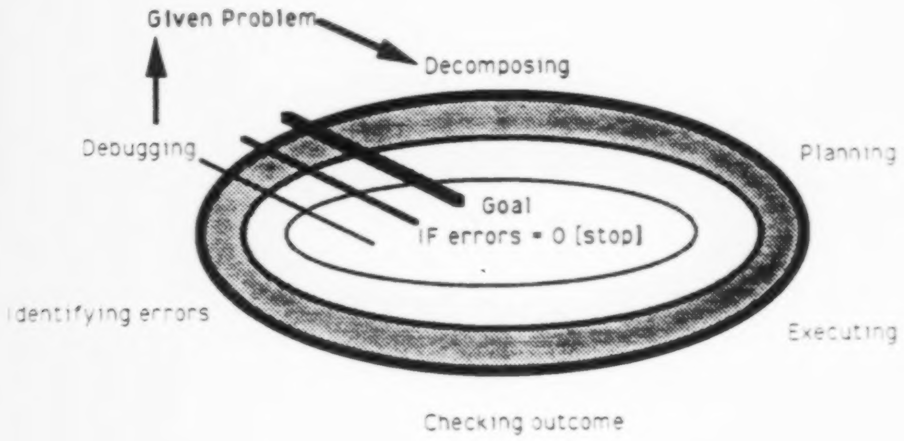


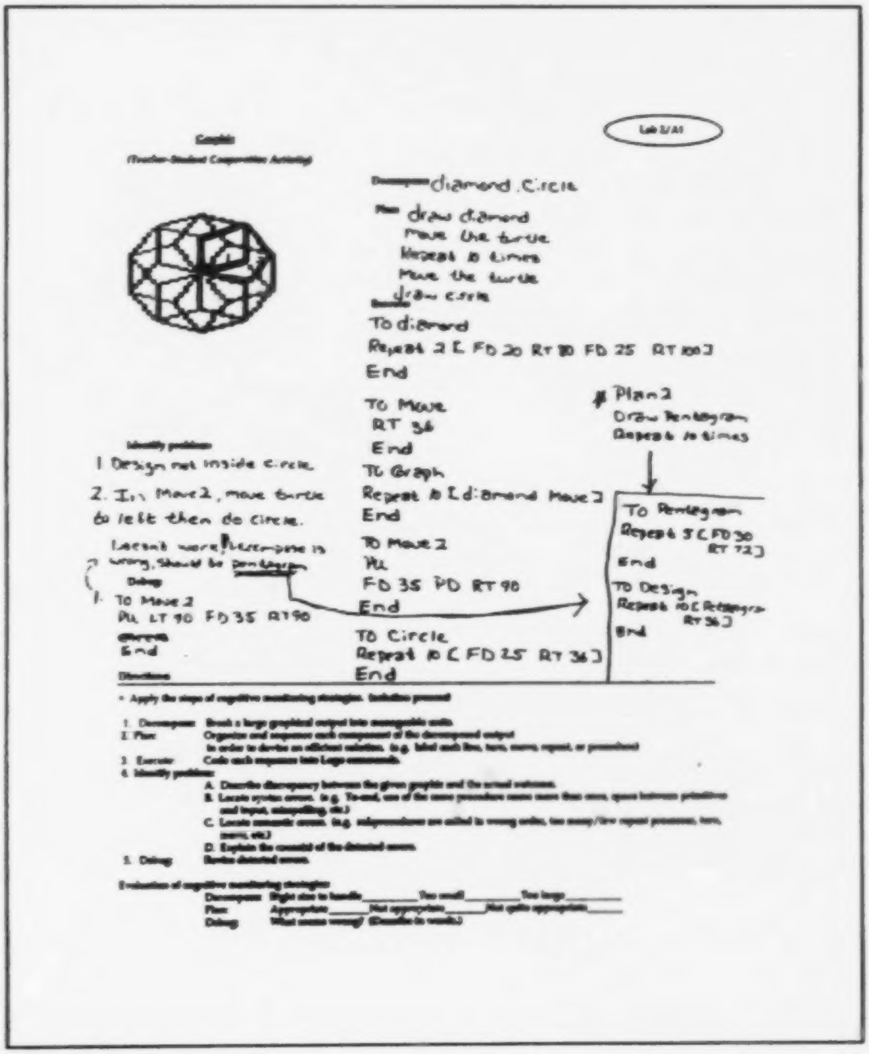
Figure 5. Recursive Cycle of Cognitive Monitoring

Once students have executed the solution on the computer, the instructor projects the outcome to a large screen. She encourages students to follow three steps of identifying the problem; describing of discrepancy between the planned graphic and the actual outcome, locating the procedures that cause errors, and explaining the cause of errors (Figure 6). These cognitive monitoring activities are cycled recursively until the problem is solved. Once students work through steps of formal cognitive monitoring activities with the instructor's guidance, then they practice the formal cognitive monitoring activities on the student activity worksheet by themselves. Finally, the instructor presents general problem situations from various subject areas and encourages students to use the cognitive monitoring processes while they are solving the given problem (Refer to Appendix C).

As the Logo session progresses, the instructor gradually reduces her role of leading students in engaging in cognitive monitoring activities and passes out the responsibility and leadership to the students. Thus, each student gradually takes a great deal of responsibility for his/her learning. The most important goals of this guided instruction are that the formal cognitive monitoring model and teacher mediated practice help students gradually internalize the learned strategies and develop the ability to automatically employ them whenever necessary.

Assessment Instruments

To assess students' learning of Logo, the researcher has developed tests which require students to demonstrate decomposing, planning, problem identification, and debugging in the



diamond, Circle

draw diamond
Move the turtle
Repeat 10 times
Move the turtle
draw circle

```
To diamond
Repeat 2 [ FD 20 RT 80 FD 25 RT 100]
End

TO Move
 RT 36
 End
TO Graph
Repeat 10 [diamond Move]
End
TO Move 2
PU
FD 35 PD RT 90
End
TO Circle
Repeat 10 [FD 25 RT 36]
End
```

Plan 2
Draw Pentagram
Repeat 10 times

```
To Pentagram
Repeat 5 [FD 30
          RT 72]
End
TO Design
Repeat 10 [Pentagram
          RT 36]
End
```

1 Design not inside circle

2. In Move2, move turtle to left then do circle.

Doesn't work! Decompose is wrong, Should be pentagram

```
1. TO Move 2
   PU LT 90 FD 35 RT 90
   End
```

Figure 6. Cognitive Monitoring Sheet

BEST COPY AVAILABLE

context of Logo programming. Because students are required to apply these skills to new problems, the tests assess the near transfer effect of guided Logo instruction. To assess students' learning of cognitive monitoring strategies associated with non-Logo areas, the researcher has developed tests which require students to demonstrate decomposing, planning, problem identification, and debugging in everyday problem situations. These tests assess the far transfer effect of guided instruction.

In the process of developing tests, each instrument was constructed by reviewing the literature in the relevant areas and generating objectives and items from literature. For example, in order to develop the general planning test, the researcher conducted a literature review in the area of planning strategies. The main literature review included Goldin & Hayes-Roth (1980), NCTM (1988), Pea & Kurland (1984), Rebok (1989), etc. Then, using telephone discussions, the researcher consulted the authors to develop test items in the context of the research theme. After the consultation, some authors provided the researcher detailed research materials relevant to this work. Following this consultation, the researcher also had discussions on the potential test items with professors in computer science, psychology, mathematics, and teacher education, in order to confirm the content validity of the test items. Several informal pilot studies helped the researcher modify and refine the test. Contents of a planning test included meal planning, systematic chemical mixing process, and daily errands (Appendix F). Then, the refined test was piloted with the subjects in the same course in Spring and Summer, 1990. In the same way, the other instruments were also developed: a Logo decomposing and planning test; a Logo problem identification and debugging test; a general decomposing test; and a general problem identification and correction test.

The Logo decomposing and planning test evaluates students' ability to break a complex Logo problem into critical units and to plan a solution to the problem. The Logo problem identification and debugging test measures students' problem identification skills and debugging skills in Logo. The purpose of the planning strategy test is to evaluate the ability to organize given information, order the sequence of actions to be performed, and think logically in order to achieve the goal most efficiently. The decomposing test measures students' ability to break a complex problem into manageable parts and correctly construct a solution using these parts. The problem identification and correction test measures students' ability to detect what is wrong with a given problem, find errors in a given solution process, explain the problem in the errors, and correct the indicated errors.

A Logo comprehension test developed by a previous researcher (Grandgenett, 1989) at Iowa State University is used to measure domain specific task performance in Logo programming.

Instructional Components

The Logo instruction consists of the following components.

1. Instructor materials: A detailed description of content, rationale, and instructional method which apply the five steps of cognitive monitoring activities in Logo problems, teacher-student cooperative activity transparencies, and examples of everyday problem transparencies to practice domain specific cognitive monitoring strategies in a general problem situation are included.
2. Student materials: Student activity sheets of Logo problems, homework criteria sheets, homework examples, and cognitive monitoring activity sheets are included.
3. Computer software: A LogoWriter disk is used to teach Logo programming.
4. Assessment instruments: In order to measure the development of cognitive monitoring strategies, a set of tests can be administered; a Logo decomposing and planning test, a Logo problem identification and debugging test, a general planning test, a general decomposing test, and a general problem identification and correction test. A Logo comprehension test is also administered to measure students' ability in Logo programming.

Empirical Evidence Supporting Guided Logo Instruction

In the Fall of 1990, an empirical study was conducted to determine the effects of guided Logo instruction on the development of cognitive monitoring strategies. The subjects were 132 students enrolled a three credit computer application course for pre-service teachers. One group used the guided Logo instruction described in this paper and the other group used a self-discovery approach to learning Logo. The two groups were compared on the following dependent variables: Logo decomposing skills, Logo planning skills, Logo error identification skills, general decomposing skills, general planning skills, and general error identification skills.

Multivariate analysis of covariance was used to analyze the hypothesis on the effectiveness of the treatment on the combined dependent variables. The result indicated that there was a statistically significant difference between the two groups on the combined means for the six dependent variables. Analysis of covariance was utilized to test each of the dependent variables. The results indicated that students in the guided Logo instruction group performed significantly higher than students in the self-discovery instruction on the measurements of the following skills: Logo planning skills, Logo error identification skills, general decomposing skills, general planning skills, and general error identification skills. However, there was no statistically difference between two groups on the Logo decomposing test (Lee & Thompson, 1991).

Summary

Educators are facing the fundamental issue of teaching transferrable skills to help students become independent learners and problem solvers in order to keep up with a rapidly changing technological information society. Cognitive monitoring has been considered as teachable, decontextual, and transferrable higher order thinking strategies. Teaching cognitive monitoring through guided Logo instruction as presented in this paper seems to have promising possibilities for developing students' efficient thinking skills.

References

Baker, L. (1989). Metacognition, Comprehension Monitoring, and the Adult Reader. *Educational Psychology Review, 1*(1), 3-38.

Baker, L., Brown, A. L. (1987). Metacognitive skills and reading. In P. D. Pearson (Ed.), Handb*ook of reading research.* New York, N.Y.: Longman.

Brown, A. L. (1978). Knowing when, where, and how to remember: A problem of metacognition. In Robert Glaser (Ed.), *Advances in instructional psychology,* Hillsdale, N.J.: Lawerence Erlbaum Associates.

Brown, A. L. (1983). Learning to learn how to read. In J. Langer and T. Smith-Burk (Eds.), *Reader meets author, bridging the gap: a psycholinguistic and social linguistic perspective.* Newark, N.J.: Dell.

Chipman, S. F., & Segal, J. W. (1985). Higher cognitive goals for education. In S. F. Chipman, J. W. Segal, &R. Glaser (Eds.), *Thinking and learning skills, Vol. 2: Research and Open Questions,* Hillsdale, NJ: Erlbaum.

Clements, D. H. (1987). Longitudinal study of the effects of Logo programming on cognitive abilities and achievement. *Journal of Educational Computing Research, 3*(1), 73-94.

Clements, D. H. (1990). Metacomponential development in a Logo programming environment. *Journal of Educational Psychology, 82*(1), 141-149.

Derry, S. J. (1989). Strategy and expertise in solving word problems. In C.B. McCormick, G.E. Miller, & M. Pressley (Eds.), *Cognitive strategy research,* New York, NY: Springer-Verlag.

Flavell, J. H. (1981). Cognitive monitoring. In W. P. Dickson (Ed.), *Children's oral communication skills* , New York, N. Y.: Academic Press.

Ford, N. (1981). recent approaches to the study and teaching of "effective learning" in higher education. *Review of Educational Research,* 51(3), 345-377.

Forrest-Pressley, D. L., Waller, T. G. (1984). *Cognition, Metacognition, and Reading.* New York, N.Y.: Springer-Verlag.

Goldin, S. E., & Hayes-Roth, B. (1980). *Individual differences in planning processes.* Rand Note: N-1488-ONR.

Grandgenett, N. F. (1989). An investigation of the potential of guided Logo programming instruction for use in the development and transfer of analogical reasoning. *Unpublished doctoral dissertation.* Iowa State University.

Lee, M. O., & Thompson, A. (1991). Effects of guided Logo programming instruction on the development of cognitive monitoring strategies. Paper to be presented at the AERA, Chicago, IL.

Leron, U. (1985). Logo today: vision and reality. *The Computing Teacher, 12*(5), 26-32.

Littlefield, J., Delclos, V., Lever, S., Clayton, K. N., Bransford, J. D., & Frank, J. J. (1988). Learning Logo: Method of Teaching, Transfer of General Skills, and Attitudes Toward School and Computers. In R. E. Mayer (Ed.), *Teaching and learning computer programming: multiple research perspectives.* Hillsdale, N.J.: Erlbaum.

Lochhead, J. (1989). Toward a scientific practice of mathematics education. *Paper presented at AERA conference,* ED 294747, 1-28.

Maddux, C. D., & Johnson, D. L. (1988). *Logo: methods and curriculum for teachers,* New York, N.Y.: Haworth Press.

National Council of Teachers of Mathematics. (1988). Mathematics Teachers. *NCTM,* 81(1), 7-43.

Papert, S. (1980). *Mindstorms: children, computers, and powerful ideas,* New York, NY: Basic Book.

Pea, R. D., & Kurland, D. M. (1984). *Logo programming and the development of planning skills.* Technical Report No. 16. New York: Bank Street College of Education, Center for Children and Technology.

Rebok, G. W. (1989). Plan, Actions, and Transactions in Solving Everyday Problems. In J. D. Sinnott (Ed.), *Everyday problem solving: theory and applications,* New York, NY: Praeger Publishers.

Salomon, G., & Perkins, D. (1987). Transfer of cognitive skills from programming: When and how. *Journal of Educational Computing Research, 3*(2), 149-169.

Seidman, R. H. (1987). *Research on teaching and learning computer programming symposium.* Paper presented to AERA, Washington, DC. ED 287 442.

Shoenfeld, A.H. (1985). *Mathematical problem solving.* New York, N.Y.: Academic press.

Singley, M. K., Anderson, J. R. (1989). *The transfer of cognitive skill.* Cambridge, Massachusetts: Harvard University Press.

Sternberg, R. J. (1984). A contextualist view of the nature of intelligence *International Journal of Psychology,* 19, 307-334.

Sternberg, R. J. (1987). Teaching intelligence: The application of cognitive psychology to the improvement of intellectual skills. In J. B. Baron & R. J. Sternberg (Eds.), *Teaching thinking skills: Theory and practice.* New York, N.Y.: W. H. Freeman and Company.

Swan, K., Black, J. B. (1989). Logo programming and the teaching and learning of problem solving. *CCT Report 89-1,* Columbia University.

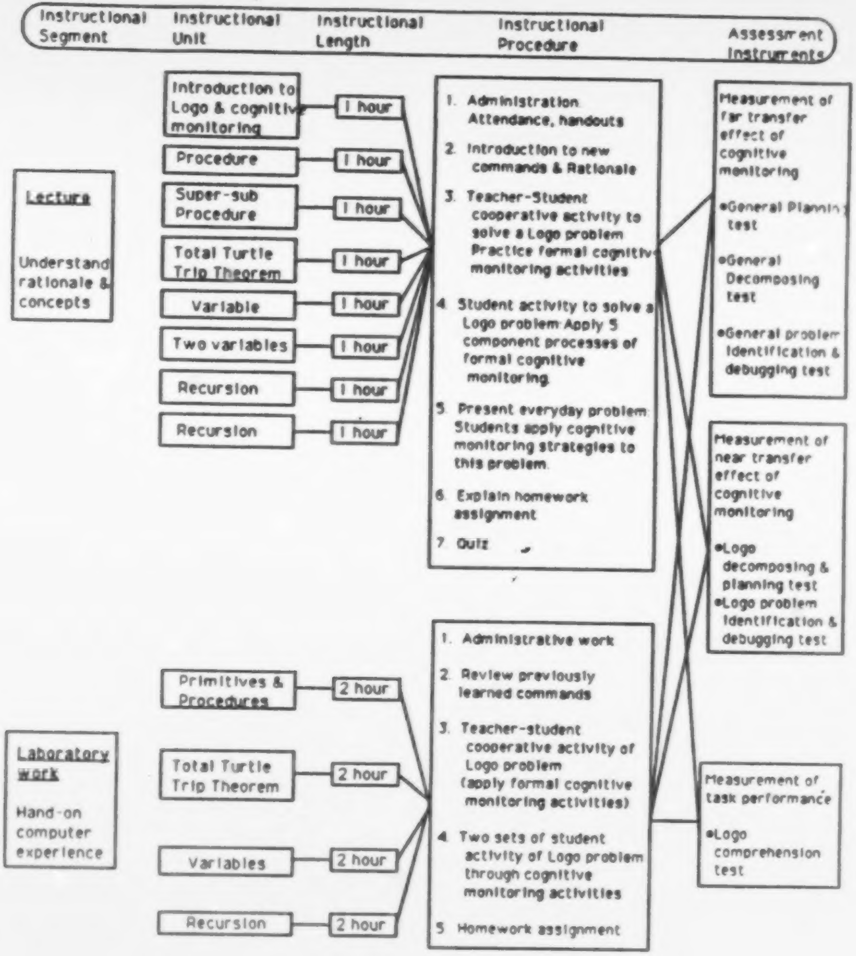
Task Force on Teaching As a Profession. (1986). *A Nation Prepared: Teachers for the 21st Century.* Carnegie forum on Education and the Economy.

Wagner, R. K., & Sternberg, R. J. (1987). Executive control in reading comprehension. In B. Britton & S. M. Glynn (Eds.), *Executive control processes in reading*. Hillsdale, NJ: Lawrence Erlbaum Associates.

Watt, M. (1982). What is Logo? *Creative Computing*, *8*(10),

Appendices to this article appear on the the following pages.

Appendix A. Summary of Guided Logo Instructional Procedure


Instructional Segment
Instructional Unit
Instructional Length
Instructional Procedure
Assessment Instruments
Lecture
Understand rationale & concepts
Introduction to Logo & cognitive monitoring
Procedure
Super-sub Procedure
Total Turtle Trip Theorem
Variable
Two variables
Recursion
Recursion
1 hour
1 hour
1 hour
1 hour
1 hour
1 hour
1 hour
1 hour
1. Administration. Attendance, handouts
2. Introduction to new commands & Rationale
3. Teacher-Student cooperative activity to solve a Logo problem. Practice formal cognitive monitoring activities
4. Student activity to solve a Logo problem. Apply 5 component processes of formal cognitive monitoring.
5. Present everyday problem. Students apply cognitive monitoring strategies to this problem.
6. Explain homework assignment
7. Quiz
Measurement of far transfer effect of cognitive monitoring
•General Planning test
•General Decomposing test
•General problem identification & debugging test
Measurement of near transfer effect of cognitive monitoring
•Logo decomposing & planning test
•Logo problem identification & debugging test
Laboratory work
Hand-on computer experience
Primitives & Procedures
Total Turtle Trip Theorem
Variables
Recursion
2 hour
2 hour
2 hour
2 hour
1. Administrative work
2. Review previously learned commands
3. Teacher-student cooperative activity of Logo problem (apply formal cognitive monitoring activities)
4. Two sets of student activity of Logo problem through cognitive monitoring activities
5. Homework assignment
Measurement of task performance
•Logo comprehension test

Appendix B: An example of a computer trouble shooting problem

Situation:

You are employed as a salesman in a microcomputer production company. Even though you have used some software, you barely know anything about the hardware. Today, your boss brought in computer system and asked you to test if it works.

Your boss does not know that you've never troubleshot the computer hardware. He seemed to trust your knowledge about the computer. You don't want to be fired since recently you borrowed a lot of money to buy a new car.

Of course, you do not want to let your boss know your problem with the hardware. You have to hide your anxiety and try to do your best in this situation. How will you handle it?

What do cognitive monitoring activities have to do with this terrible problem situation? Utilizing cognitive monitoring strategies may help you solve this problem.

A desired goal: Test a new Apple computer if it works.

1. Decomposing stage:

1. Separate each part of the computer and understand its function.
2. Parts of the computer are: Computer, Keyboard, disk drive, monitor, cables and software.
3. Cables are used to connect each part of the computer.

2. Planning stage:

1. Need to figure out which cables are to be connected to which parts of the computer.
2. Should connect a disk drive, a monitor, and a keyboard to the computer.
3. Need software to test if the computer works after connecting all the parts of computer.

3. Executing stage:

1. Spread all the parts of the computer out on the desk.
2. Actually connect disk drives, monitor, and keyboard to the computer.
3. Insert a disk into the disk drive and close the disk drive door.
4. Turn on the computer power switch.

Actual outcome:

The red light on the disk drive flashes on and off, but nothing appears on the screen.

4. Identifying problem stage:

Since nothing appears on the screen, check the monitor first.

1. Was the monitor properly connected to the computer?
2. Was the monitor's switch turned on?
3. How about the screen contrast?

5. Debugging stage:

1. Connect the monitor cable to the computer tightly.
2. Turn on the monitor switch.
3. Adjust the screen contrast.

Appendix C: Application of Cognitive Monitoring to Build A House

The Rittscher family is building a new house. Scott is an electrician and Pole is a plumber. Jill, Kerri, Pat, and Tom are good hard workers and good with their hands. So they are going to do most of the work themselves.

Here is a list of tasks, the amount of time needed to complete each task, and the order in which the tasks must be completed..

Task	*Time*	*Preceding Steps*
A. Site preparation	**4 days**	**none**
B. Foundation	**6 days**	**site preparation (A)**
C. Drains and services	**3 days**	**site preparation (A)**
D. Framing	**10 days**	**foundation (B)**
E. Roof	**5 days**	**framing (D)**
F. Windows	**2 days**	**roof (E)**
G. Plumbing	**4 days**	**drains and services C)** **roof (E)**
H. Electrical work	**3 days**	**roof (E)**
I. Insulation	**2 days**	**plumbing (G)** **electrical work (H)**
J. Shell	**6 days**	**windows (F)**
K. Cleanup &paint	**6 days**	**insulation (I)**

Question 1: On what day will Scott and Pole be needed?

Question 2: Jill and Kerri will only be able to help during cleanup and painting. Until then they will be with their grandparents. How long will their vacation be?

Understanding the problem and Decomposing:

1. Site preparation, foundation, framing, and roof needs to be completed before plumbing and electrical work can begin.
2. The electrician can start the same day as the plumber.
3. Site preparation, foundation, framing, roof, windows, and shell need to be completed before cleanup and painting can begin.

Planning:

1. Sequences
2. Strategies: draw a diagram

Executing:

1. Plumbing is G, therefore, A(4)+B(6)+D(10)+E(5)=25 days.
2. Electrical work is H. The exact same jobs must be done before the electrician can start. Therefore, both the plumber and the electrician can start on the 26th day.
3. Cleanup and painting is job K. The fastest time the shell (J) can be completed in is: A(4)+B(6)+D(10)+E(5)+G(4)+I(2)=31 days

Identifying problem and debugging:

Are the answers right? Let students think through and check the answers.
If the answers are not right, then let them find what is seemingly wrong.

Appendix D: Lecture Day 5-Lesson Outline

Pass around the lecture attendance sheets.

Announcements:

1) Remind students that their quizzes will be returned during the LogoWriter lab.
2) Logo midterm will be Tuesday October, 16.
3) Others ______________________

Review Names of Procedures

- Remind students that they need to use different names of procedures for a different direction of moves, a different orientation of graphics, and so on: *Move, Move1, Move2...*

Introduce Variables

- *Rationale of learning variables*
  1. Utilizing specific information for a general purpose is important to solving various problems successfully.
     - Practice looking for a general solution to a specific problem so that it can be applied to new and different problem situations.
  2. Variable is an important idea in mathematics (abstract, general).
  3. Variable allows students to manipulate expressions.

On Computer:

- Briefly show the square procedure:

```
To Square
Repeat 4 [Fd 50 Rt 90]
End
```

Discuss that if we want a square of a different size, we need to go in and actually change the procedure or retype different name.

```
To Square :X
Repeat 4 [Fd :X Rt 90]
```

Show that this procedure is much more powerful and flexible.

- Mention that the name of the variable can be anything but numbers. However, a letter followed by a number is acceptable (No space between a letter and a number).

Activity Sheet 1 (Pass out sheet) :Teacher-Student Cooperative Work

Transparency1:

- The instructor leads students in applying the steps of cognitive monitoring strategies on the sheet. She uses Socratic dialogue in order for students to focus on their thinking process deliberately. **(Decompose, Plan, Execute)**

On Computer:

- Show the result.
- **Identify Problem:**
  1. Locate the code(s) that causes bugs.
  2. Discuss what the problem is in the code(s).
- **Debug:** 3. Correct the code.

Activity Sheet 2 (Pass out sheet) : Students work individually

Transparency2:

- Allow students to apply the steps of cognitive monitoring strategies for about 5 to 10 minutes. **(Decompose, Plan, Execute)**
- Gather a common solution and write it on the transparency.

On Computer:

- Show a student example of the solution.
- **Identify Problem:**
  1. Locate the code(s) that causes bugs.
  2. Discuss what the problem is in the code(s).
- **Debug:** 3. Correct the code.
- Ask students if they have a more efficient solution.
- Emphasize the syntax format for variable procedures.

Relate Cognitive Monitoring to Building a House

Transparency3:

- Discuss how to utilize cognitive monitoring components in a given problem solving situation.

Mention Homework Assignment #3

- Mention that homework assignment #3 will be similar to the previous projects, but will use variables.
- More details will be discussed in the next lecture and lab.

Distribute Mini-Quiz (if time allows)

Laboratory work 2 -Lesson Outline

Attendance, Record Keeping, Announcements

- Hand back any assignments that need to be returned to students.
- Have students turn in their LogoWriter projects by:
  1) Booting up their project so that it can appear on the screen.
  2) Making sure that the page holding their project is named with their "**lastname1**", if not, they need to rename the page with this name.
  3) Inserting the instructor's master disk, and pressing <escape>. (This saves it on the instructor's disk.)
  4) Students must turn in their planning and criteria sheets but keep their own LogoWriter disk.

Review Regular Polygon *MONITORS OFF*

Demonstration: •Briefly show the total turtle trip theorem.

```
Repeat __[Fd __ Rt __] or     (Total turtle trip should be 360 degrees)
Repeat __[Fd __ Lt __]
```

•Have students figure out **degrees** and **shapes**.

```
Repeat 6[Fd 40 Rt __]     (Hexagon)
Repeat 5[Fd 40 Lt __]     (Pentagon)
```

•Briefly show the following procedure:

```
To Stick              To Line
Fd 40 Bk 40           Repeat 2[Stick Rt 360/2]
End                   End
```

MONITORS ON

On Computers: •Have students type the above procedure.
•Then, have students make *CROSS, ASTERISK, SNOWFLAKE*, .etc.

```
To Cross                   To Asterisk
Repeat 4[Stick Rt 90]      Repeat 6[Stick Rt 360/6]
End                        End

To Snowflake
Repeat 8[Stick Rt 360/8]
End
```

MONITORS OFF

Cognitive Monitoring Activity Sheet #Lab2/A1 (pass out the sheets now.)

Transparency1: •Work through the sheet of cognitive monitoring strategies step by step with student discussion.
(DECOMPOSE, PLAN, EXECUTE) *Students not on computer*

On Computer: •Execute the plan and show the result.
•*IDENTIFY PROBLEM:*
1. Discuss what is the discrepancy.
2. Discuss the location of code(s) that causes error.
3. Discuss what the problem is in the code(s).

•*DEBUG:* Correct the code(s) and see the new result.

MONITORS OFF

Cognitive Monitoring Activity Sheet #Lab2/A2 (pass out the sheets now.)

Transparency2: •Allow students to apply the steps of cognitive monitoring strategies for about 10 minutes.
(DECOMPOSE, PLAN, EXECUTE)

MONITORS ON

On Computers: •Have students try their program on the computer.

On the Sheets: •*IDENTIFY PROBLEM:*

1. Discuss what is the discrepancy.
2. Discuss the location of code(s) that causes error.
3. Discuss what the problem is in the code(s).

On Computers: •*DEBUG:* Correct the code(s) and see the new result.

•*INSTRUCTOR HELPS STUDENTS INDIVIDUALLY.*

FEEL FREE TO TALK ABOUT THEIR PROJECTS AND ALLOW STUDENTS TO COOPERATE WITH EACH OTHER TO CORRECT ERRORS.

Review Homework Assignment #2

•Distribute the homework planning sheets and criteria sheets.

Transparency: •Discuss what is expected for Homework assignment #2 by going over the planning and criteria sheets.

STUDENTS MUST HAND IN A COMPLETED PLANNING SHEET WITH THE PROJECT NEXT WEEK.

Make sure students save at least once while in the lab

Allow students to work on Homework

•For the rest of the time, students work on their own homework.

•Students should leave lab with at least part of their project saved under a page named with "lastname2".

Appendix E. Examples of Socratic dialogue to elicit cognitive monitoring activities.

- *Decomposing process:* "What shall we do with this problem?", "Shall we write codes now? if yes, what do we know about this problem?", "What are the important elements of this problem?"
- *Planning process:* "How would you relate the decomposed elements to solve the problem?", "Where would be the best place for the turtle to start to travel?", "Can we organize gathered information in more efficient ways?", "What strategies would you use?"
- *Executing process:* "How does the turtle understand our plan so that he can carry it out ?"
- *Identifying problem process:* "Ooh, what happened?", "Can you describe the discrepancy between our planned graphic and the actual outcome?", "Where might the errors be?", "Did our plan seem O.K.?", "Did our decomposed elements seem fine?", "Can you describe what is wrong with the procedure you mentioned?"
- *Debugging process:* "What would be the correct codes?", "How can you correct our errors?"

Appendix F. A planning test

Name: ______________________________

Lab Section: _______________

Planning Strategy Test

The purpose of this test is to evaluate students' ability to organize given information, to order the sequence of actions to be performed, and think logically in order to achieve the goal most efficiently.

Directions:

Read the given problem carefully, and specifically describe your solution processes to the problem.

Try to seek cues or hints in a given problem.

I know I can do it.....

Problem 1.

Tom's parents are coming to visit him in his new apartment. He wants to plan a special meal and have *everything done at the same time*. Here is a list of the items he will be preparing and the amount of time it will take for each of those items to be completed. Help Tom out by telling when he should start each task.

Bake potatoes	*1 1/2 hours*
Cook the roast (in a crock pot)	*6 hours*
Make a salad	*10 minutes*
Set the table	*8 minutes*
Put dinner rolls into a basket	*1 minutes*
Cook dinner rolls	*18 minutes*

Some things can be done at any time. But he wants to wait *as late as possible* to do each of them. Dinner is going to be served at 6:30pm.

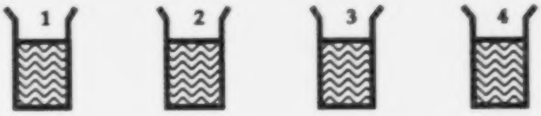
Your plan to help him prepare dinner is:

Things to do	*Time Tom would start*

Problem 2.

There are 4 large bottles each filled to the same level with different fluids that all look like plain water. The bottles are distinguished only by the numbers, 1 through 4. Only one combination of these fluids will make a bottle of pretty yellow-colored fluid (It could be any number of fluids). Since there is no label for the fluids, you have to experiment to make a bottle of yellow-colored fluid. You cannot use any other fluids.

Write down all the possible combinations you would systematically try for identifying the correct combination which make a bottle of yellow-colored fluid.



Problem 3.

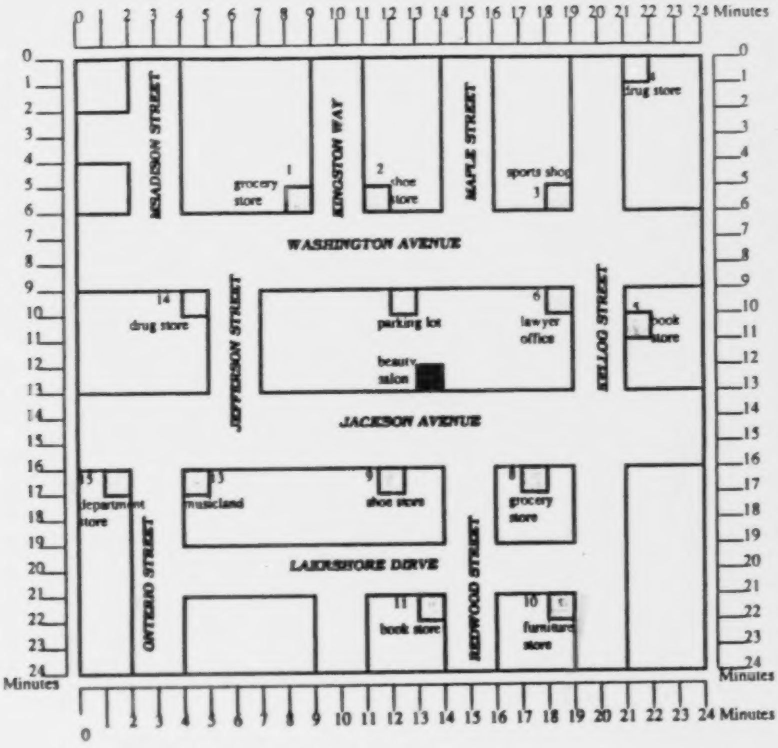
It's now **9:30 a.m.** and you will be finished with your haircut appointment in **10 minutes**. Now, you are planning today's errands and have to do *as many things on your list as possible* **before being at the Washington Avenue parking lot by 12:00 noon** (refer to the attached map).

You are traveling by foot and it takes 25 minutes to cross town in either direction. Here's your list of errands. You are free to do them in any order you like.

Errand List	**Shop #**	**Expected Spending Time**
Buy CD's at musicland	13	15 minutes
Buy shoes at one of the shoe stores	2, 9	15 minutes
Fill prescription at one of the drug stores	4, 14	15 minutes
Sign papers at lawyer's office (due today)	6	20 minutes
Look at furniture at furniture store	10	15 minutes
Buy required books for tonight's art history class at one of the book stores	11, 5	15 minutes
Buy fresh vegetables and ice cream at one of the grocery stores (will use these tomorrow morning)	1, 8	15 minutes
Buy tennis racket at sports equipment store for this afternoon's class	3	20 minutes
Buy a sweatshirt at department store	15	20 minutes

• **Write down your plan to do errands.** You can sketch time and distance on the back of this paper or on the map before fill out the following categories.

Name of Errands you will do	**Shop # you will go**	**Time you will spend**
• **Being at the Washington Street Parking lot by 12:00 noon.**		


Minutes
MADISON STREET
KINGSTON WAY
MAPLE STREET
1
grocery store
2
shoe store
sports shop
3
4
drug store
WASHINGTON AVENUE
14
drug store
JEFFERSON STREET
parking lot
6
lawyer office
KELLOG STREET
5
book store
beauty salon
JACKSON AVENUE
15
department store
13
musicland
9
shoe store
8
grocery store
ONTARIO STREET
LAKESHORE DRIVE
REDWOOD STREET
11
book store
10
furniture store

Projects
Looking into the Future

Increasing Technological Skills through the Training of Teacher Leaders

Madeleine Long
Long Island University, New York

Abstract

Abstract unavailable at press time.

Taking the *Next Steps...*

Martha Hancock
Jefferson County Public Schools, Louisville, KY

Abstract

Next Steps... is a comprehensive plan for increasing the use of instructional technology in Jefferson County Public Schools in Louisville, KY. It seeks to provide a framework by which all schools, teachers, and children will continue to systematically infuse technology into the learning process. The plan addresses the need for technology management in three levels: a steering committee, site-based management teams, and centralized services. This presentation will discuss the site-based management teams. These site-based teams are currently in process of developing a comprehensive technology use plan, which takes into account not only new technologies, but better use of existing technology. The plans are being derived from an intensive self-study done in each school. The self-study included the following criteria:

- an assessment of the instructional needs of the student population;
- a definition of learning outcomes desired;
- an identification of the student population who will use the specific applications;
- a plan for use of existing resources;
- a design for community use;
- evaluation and assessment criteria;
- a staff and curriculum development plan;
- staff and student organization;
- priorities for implementation;
- a home/school computer plan;
- a technical education plan; and
- a computer ethics commitment.

By attending this presentation, participants will gain knowledge in setting up a site-based technology committee, learn how to create a self-study, and obtain the structure needed for technology planning.

Virginia's Plan: A Blueprint for the Present and the Future

Helen Edens
Virginia Department of Education

Abstract

Abstract unavailable at press time.

Projects
Special Learners

Integrating Computers into a Communication Disorders Curriculum

Paula Cochran
New Mexico State University

Abstract

Abstract unavailable at press time.

Use of the Macintosh as an Augmentative Communication Device

Ray Braswell
Auburn University at Montgomery

Abstract

This project is based upon the use of the Macintosh computer, MacRecorder, and a touch screen to provide a digitized speech communication system for use by those persons who cannot speak. A prototype of this system is presently being developed and will be field tested upon completion.

With the capabilities of the system, digitized speech can be adapted for the intended user. If the user is a 10-year-old female, the voice utilized can be that of a female child. If the user is a 35-year-old male, the voice utilized can be that of an adult male. The vocabulary requirements can be adapted or increased according to need or specific requirements. In addition, the voice utilized can be an actual human voice, as compared to the device utilized by many present systems which prevents the "speaking voice" as a fairly monotone, mechanical-sounding synthesized voice.

The addition of a touch screen to the Macintosh allows for those users with limited movement to utilize the system. The area of the screen to be touched can be adapted for various users with a wide range of physical abilities.

Computer Access for Visually Impaired College Students

Janet Smith
University of Tennessee

Abstract

The percentage of first year college students with disabilities increased from 2.6% in 1978 to 7.4% in 1985. The numbers continue to increase, especially since the enactment of the Americans with Disabilities Act. As enrollment has increased, there has been increased concern about the programs and services that can best meet the needs of college students with disabilities. Availability of computer technology has been identified as a potentially important student service that can enhance the quality of overall college experience for students with disabilities.

The author of this proposal received some internal funding from the university to purchase a few pieces of adaptive computer equipment (ACE) for visually impaired students. The main objective of the project was to introduce the university community to ACE and the opportunities it can provide to students with disabilities.

Projects
Global Classroom Using Telecommunications

Teleconnected Cultures: An International Telecomputing Classroom Project

Bonnie Price
Alhambra School District, Alhambra, CA

Abstract

This project pairs classrooms from various locations to compare cultures with each other through telecomputing. Students in each class first research their family culture, share within the classroom, then send letters written on the computer to students in their paired class over the modem. After receiving the first messages from the paired class, the students then research the culture of their school and community, formulate more messages to send through telecomputing, and compile a set of artifacts from the school culture to send through the "regular" mail to the other class.

The curriculum for this project was developed as an action plan to serve as an outcome of participation in the Christa McAuliffe Institute, a project of the National Foundation for the Improvement of Education. The project has been conducted on three telecommunications systems, McGraw Hill Information Exchange (no longer in operation), Iris Online Network for Teachers and Schools, and the California State University Network (CSUnet). Over the past three years of conducting this project, more than 70 classes have participated in this project, from fifteen states and four countries. During the 1990-91 school year, the project has served a total of 42 classrooms. The class sizes ranged from 8 to 30, averaging about 23.

Teleconnected Cultures provides students with the opportunity to share very real aspects of their daily lives with students in other locations. This has the effect of opening the classroom walls to embrace the outer world, making the study of culture and geography much more personal and relevant. Student interest in other cultures helps bridge cultural gaps that separate students of different backgrounds, creating a global classroom to prepare children to live in a global community.

A Telecommunication Tale of War-Time Wonders

*Gideon Goldstein, I*EARN Center Coordinator at ORT Israel*
Jeffrey Golub, Shorecrest High School, Seattle, Washington

Abstract

Several telecommunication projects were initiated during the past years. Common to all was the careful structuring of both curricula and teaching methods.

The real test of a medium is its durability in impromptu conditions. The same applies for teaching methods. The project described will examine electronic conferencing and distance learning as well as the viability of telecommunications in an improvised situation.

The Desert Storm that blew over the Middle East this winter carried in its wake an examplary project in teleconferencing and distant tutoring. As part of the International Education and Research Network (I*EARN), founded and supported by the Copen Family Fund, a special electronic conference was set up at the onset of the war. The conference enabled studjents from both the U.S. and Israel to engage in daily dialogue about the emotional and political impact of the war. The conference was initially facilitated by a teacher but was soon handed over for the students to run. This electronic connection also allowed students from Shorecrest High School in Seattle, Washington to engage in a Middle East research project supervised by an Israeli educator over 10,000 miles away. The research, covering several facets of the disputes in the region, involved library work, interviews and data collection. Every few days the students would provide their progress to their "distant" teacher, who would, after assessment, assign new tasks.

The success of the conference and the electronic correspondence indicates that other long-distance learning projects should be encouraged. The concept of the "global classroom" is an idea worth pursuing and developing further.

Special Presentation
Learning Research Institute

Ted Kraver
President, Kumm Industries
Phoenix, Arizona

Abstract unavailable at press time.

Special Presentation
Exemplary and Typical Computer-Using Teachers

Data are presented from two national surveys of computer-using teachers—one, a survey of 600 reputationally expert computer-users conducted by Bank St. College of Education and the other, a national probability sample survey of 3,000 teachers conducted by Johns Hopkins University as part of the international I.E.A. Comp-Ed survey. Suggestions made about the differences in environments and perspectives between these two groups of teachers help us to focus on the problem of extending "best practices" to teachers with less computer experience and to teachers who might have contrary perspectives on the teaching role and the most important educational objectives.

Presenters:

Henry Jay Becker
Center for Social Organization of Schools
The Johns Hopkins University

Karen Sheingold
Educational Testing Service

Martha Hadley
Center for Technology in Education
Bank St. College of Education

Special Presentation
Beyond Multimedia to the Virtual Classroom

Fred D'Ignazio
Multi-Media Classrooms, Inc.,
East Lansing, Michigan

Few teachers feel comfortable teaching using the new tools of technology, including computers, video cameras, laserdiscs, and telecommunications. Those who do use these technologies often proceed cautiously and never get a glimpse of the exciting ways these tools can help a teacher implement new *paradigms* for classroom learning. The Teacher Explorer Center is a teacher-training facility funded by the Michigan State Board of Education and co-sponsored by 30 business partners. At the Teacher Explorer Center we are conducting workshops for over four thousand teachers and administrators representing more than a quarter of Michigan's school districts. These workshops *link* the new multimedia learning technologies with new teaching strategies, including whole-language publishing, thematic inquiry, and cooperative learning. This presentation will introduce the audience to the Teacher Explorer Center and will demonstrate how teachers can create an exciting new environment for teaching that encourages ingenuity, imagination, pride of authorship, joy of learning, and intense student participation.

Paper

Technology and Teacher Autonomy: Findings of an In-School Study

Peter Gray Whiteley
Hamline University, Minnesota

Abstract

This paper reports findings on teacher autonomy. After discussing the overall context of computer and information based technology-in-education [CIBT] efforts, the author draws on qualitative evaluation data to construct a composite picture of the pressures on classroom teachers involved in CIBT programs, then uses teachers' responses to evaluation questions to present a teacher-perspective on these matters. The paper concludes by reflecting on the need for teacher autonomy in CIBT attempts.

Context

To raise the question of teacher autonomy is to touch on enduring issues in education and schooling. For teachers, exercising autonomy implies taking risks and responsibility for what occurs in curricula they devise. It requires exercise of imagination in what may be unusual and unconventional ways, expending creative energy, adopting an attitude of experiment and of openness to ambiguity, and growth and susceptibility to change in directions which emerge as warranted. While few might publicly object to teachers with these qualities, the record continues to show that they are generally discouraged in formal education institutions, where such autonomy seems dysfunctional [1]. Fiscal and resource administrators, personnel and curriculum officers, and most teaching staff in education institutions operate within structures and mind-sets whose limits are overwhelmingly conventional. Perhaps attractive when it implies garnering outside funds for a site, at this level the phrase "educational innovation" nonetheless raises general fears of discomfort, mismatch with tradition(s), and changed relationships.

These tendencies acquire a particular complexity and urgency in relation to computer and information based technology-in-education [CIBT] attempts. Complexity stems not just from an increasing volume of CIBT-related technology and the energy and salesmanship with which it is introduced, but also from a commensurate bewilderment which persists within the educational establishment about what CIBT is, where it came from, where it is heading, and how far to go with it. Urgency arises from the need to respond to the pedagogical and epistemological novelty claimed for CIBT by its manufacturers and promoters. Teachers simultaneously have to work within, evaluate, and interpret CIBT claims to themselves and to uncertain schools, districts, and other education constituencies. In-class, where inevitably it falls to teachers to hold a tentative and shifting middle-ground of CIBT 'innovation', reality belies much extra-class rhetoric. The classroom becomes an awkward arena predominantly for others' curriculum ideas, but one in which teachers are held accountable for 'making things work'.

As a result, though a general mood of enthusiasm may surround CIBT for those closest to it, and though discussion of CIBT issues in education usually has a positive face, how these outward dispositions are best articulated into curriculum is consequently confused [2]. CIBT promoters urging their particular interests to a bewildered education establishment, with teachers as *ad hoc* arbiters, is not a scenario for imaginative teacher-authored curriculum. Rather, under such circumstances teacher authorship, imagination, creativity, and energy are skewed away from genuine curriculum choice. Imprecise and variable ideas on all sides about what curriculum consists in and how it is to be engendered have in the past beleaguered teachers; risks associated with such lack of perspective are exacerbated in the CIBT process.

What does seem consistent in approaches taken is the assumption that CIBT curricula should involve teachers as crucial intermediaries in learner-CIBT relationships, and within this assumption the question of teacher autonomy invites further consideration. To address this, in the next two sections we shall draw a picture of the reality teachers face where CIBT programs are undertaken, and then review teacher perspectives on this reality. The 'reality' and 'perspective' offered here are composites drawn especially from ongoing evaluation of an extensive four-year CIBT attempt [3], and will be the basis for the final section's comments on autonomy.

Teacher Realities

Despite vigorous commentary, critique, debate, claims, counterclaims, and conferencing, clear indicators of what is to count as success in the adoption of CIBT in schools remain amorphous [4]. Establishing a CIBT program—of whatever scope or ambition—is by definition experimental in such a climate. To undertake a CIBT program is to bring all the usual implicates of schooling (e.g., schedules, resource allocation, expectations of personnel, approaches to pedagogy, types of outcomes, forms of evaluation) under an umbrella of re-accommodation which is often far outside locally accepted norms. Notably what 'models' there may be to guide program development—concise inferences from what others have done—are usually particularistic around certain narrow and only marginally transferable aims. Added to this, and confusingly, experimental processes' projected end-points and effects on existing programs regularly diverge from initial predictions.

Institutionally, CIBT invests teachers with new roles from the outset. In ways for which few teacher preparation programs have readied them, teachers must continually develop new levels of technical expertise while at the same time they maintain their focus as educators for whom—presumably—students rather than technology take priority. Teachers must constantly research, appraise, and reappraise the claims of a vigorous and sophisticated marketplace, reviewing and critically incorporating hardware and software in the day to day classroom. As well, they must make novel demands on and interpret their activity to each other and to parents, administrators, and members of an often uneasy school population. In all this teachers must note the relative effectiveness of what it is they are trying out, for their own and others' understanding of CIBT attempts.

From the workface perspective of the classroom, it appears that a growing population of hardware and software manufacturers assert their respective products' educational importance with little examination of the nature of curriculum or issues of curriculum choice undergirding the assertions made. Maximization of techno-expertise along certain pedagogical lines is not equivalent to maximization of the educationally sound and necessary. No technology is a 'fixit' for all curriculum woes; creative adaptation of technology to curriculum requires an educational imagination as yet not clearly demonstrated by manufacturers.

As CIBT participants, teachers have to struggle with these and a number of other factors, both internal and external. Responsibility to their subject-disciplines and their experience and understanding of children in the ecology of classrooms predispose teachers to a critical disposition toward manufacturer and institutional *desiderata*. Frequently inveigled into CIBT attempts because of their dynamism and success in conventional programs, [even] the most experienced teachers face dislocation from past reliable routines of planning and presentation. Incorporation of available technology involves an unfamiliar curriculum-bootstrapping for such teachers, as they try simultaneously to effect a detailed, ongoing, critical review of new and unique materials while offering a stable, sufficient curriculum. Among other things, longer hours, increased CIBT-related extra-class commitments (especially in the continual upgrading of their own technical skills), close scrutiny and changed expectations from colleagues and other members of the institutional population, coordination of resources in ways for which procedures are not set up or are inadequate, and reconciliation of novel outcomes to standardized formats all confront CIBT program teachers.

In the short term, professional rewards for participation in a CIBT program are unlikely to come until after an extended period of ambiguity and hard work. Teachers are especially accountable in an experimental program to students, other CIBT teachers, non-CIBT teachers, all levels of administrators, funding agencies, manufacturer's representatives, evaluators, researchers, and the curious. They are meanwhile accountable for the in-class program details of pinning technology and its attendant enthusiasms back to classroom practice. The greatest practical consideration day to day always seems to be lack of time to achieve all this.

Moreover, while factors such as enthusiasm, the attractiveness of manufacturer promotions, local pressure (e.g., from parents or community groups), or a competitive urge not to be left behind may decidedly stimulate institutions into CIBT programming, the wherefores, whys, and direction of continued institutional and therefore teacher participation in much of this programming seem far less certain. CIBT infrastructure, its hardware and software, are certainly not going to be uninvented, nor are they characterized by *stasis*. A qualitative difference obtains between CIBT and the steady cycle of materials which have in the past fallen into a 'this year's fad, next year's closet' syndrome in education. Engaging in CIBT programs implies commitment to educational aims which are more than one-shot, and in turn this implies an institutional imagination and commitment as yet hard to discern in any widespread sense. Where the long term is institutionally ill-considered or under-considered, CIBT teachers face the added uncertainty of what constructive use will be made from what they have done, as well as uncertainty about the future direction of their own careers. Given the hierarchical structure of most education institutions and the perceived place of teachers in that hierarchy, teachers are the institutional personnel most vulnerable to poorly conceived policy (*a fortiori* those teachers working in 'experimental' programs).

The Teacher Perspective

With regard to the teacher perspective on the reality outlined in the previous section, a pattern emerges. It is presented immediately below around a series of questions asked in the course of the qualitative evaluation [5] on which this paper is based. This dialogue between Interviewer [I] and Teacher [T] is presented to elucidate issues relevant to CIBT teacher autonomy.

I: From your experience with and exposure to an extensive CIBT program attempt—now in its third year—what advice would you offer others contemplating CIBT efforts?

T: It's important to be clear on whose idea it is to start with. If it's an administrator's then s/he may have an advantage in getting funding, and very likely a successful grant-writer to support implementing a program. But the people who will teach in it have to become excited to develop it. Teachers have to have a free rein to develop it where their excitement leads; an administrator might plant some seeds but has to let teachers develop a program from their perspective and then help and guide and support that.

When the program is operating teachers have to feel solid support and a sense of what and who the program is for. Their motivation is related to the empathy others feel for what they are going through. Teachers in CIBT programs will need to be treated differently because all sorts of exceptions to routines may have to be made. In school, that normally means support has to come from the highest level.

I: What about the perceptions of others in the school toward such a program?

T: CIBT teachers can expect mixed messages from others, for example, from other teachers. Teachers are very conscious of what others on staff 'get', and with CIBT the resources and budgets involved can seem very large compared to allocations per non-CIBT teacher. The different treatment in school which is often necessary for CIBT program teachers, and the school and community publicity around a CIBT program, can be interpreted by other teachers as favored treatment.

Sometimes this can translate into a reluctance by other teachers to attend meetings where the program's teaching strategies or results are being shared, and sometimes it seems other teachers translate the program's success to mean that the rest of the school will be seen to be failing somehow. While they might like you as a person other teachers can be reluctant to talk with you about what you are doing and what it might offer them or what they might offer it. That's a change with people who have been fairly close colleagues for some time. It can also be difficult to get others to discriminate between the effect of the technology and of non-technology factors in the results a program gets; success is usually attributed only to the resources you have that they don't.

As well, some administrators who might not understand the program or who were not part of developing it might also resist the amounts of money and equipment involved or the level of attention apparently going to a select sub-group in the school. While as a teacher you might see the program as having a long-term benefit and place (you hope it will), they might see it as some one-time thing which ceases when the current round of funds runs out.

I: What are a CIBT teacher's strongest sources of support?

T: Usually others in the instructional team with which you are working—other close members of the project.

I: What about support from technology sources? Is it forthcoming from hardware and software manufacturers?

T: Yes and no. It's not clear manufacturers are really aware of what happens in a classroom—when we talk with them at conferences they're not taking notes. They have yet to find the balance between the usefulness of their products for a teacher and that same teacher's total curriculum concern. In this sense hardware and software go hand in hand in what they require of a teacher; for the teacher it's a matter of always working with what others, manufacturers, devise for you to use and always having to adapt it.

In what it offers compared to the time it takes to review and adapt to each class, much software is no better than a textbook. It's OK for demonstrations, but good one-on-one software is not available in a lot of areas. And it seems that for marketing purposes software is as genericised as textbooks, with manufacturers opting for a very noncontroversial approach. It's also presented with a uniform style and format for all units in an area. But to keep interest in a classroom things have to be much more varied.

I: What is your chief need as a teacher in a CIBT program?

T: Time. More time to sort out how we can take what's so far available and make it fit with what's coming and with what we'd like and how all of that meets the state and school requirements we still have to work under.

Initially, trying to do all this was a night by night, weekend by weekend process of survival. It has been easier after the first year, now that some routines have been established and we have a better sense of the technology and its fit with the intellectual, social, and emotional levels of the students. But we still average more than 50 hours a week of program commitment from each teacher during the semester, with an extra two months student-free for review and planning in summer.

So much of our time is spent in meetings called to coordinate requirements from—or to meet with—members of the school administration, interested visitors, colleges, manufacturers, media, or the counsellor, as well as for integrating our curriculum planning. There's always someone coming through the door.

As teachers we need time for ourselves, to take a step back and try to get a sense of where we are going, where this energy we're putting out will lead for us, to try to sort through the demands of the program to see which ones are meaningful for us. There's not enough of that.

I: Do you see time becoming less of an issue?

T: There's never enough time to review even half of what's out there and still make things work here in a full-fledged way.

I: What ownership does a teacher have in a program?

T: You're identified with it and are responsible to everyone for it.

With reference to this interchange and the material of earlier sections, we shall now seek what preliminary conclusions may be drawn about teacher autonomy in CIBT attempts.

Teacher Autonomy

In terms of the attributes of an autonomous teacher mentioned above (i.e., taking risks and responsibility for curricula they devise, exercise of imagination in unusual and unconventional ways, output of creative energy, experiment and openness to ambiguity, growth and change as warranted) it seems that within CIBT efforts teachers are severely checked before they begin. Responsibility seems to be theirs—'You are responsible to everyone'—but not for curricula over which they have reasonable control. Note the comments that 'it's a matter of always working with what others devise for you to use and always having to adapt it', and that 'while as a teacher you might see the program as having a long-term benefit and place (you hope it will), administrators might see it as some one-time thing'; teachers' own devices and even sense of direction are rendered vicarious. Further, a chief need is 'more time to sort out how we can take what's so far available and make it fit with what's coming and with what we'd like and how all of that meets the state and school requirements we still have to work under'. Without this time, curriculum remains constantly unsettled as the school, manufacturers, the community and the state are all impacting on the classroom for different reasons, none of which are *a priori* meaningful from the in-class teacher's point of view.

Imagination does seem exercised and energy is spent, but in a draining, always-having-to-adapt way for which institutional support may or may not be forthcoming. If for no other reason than this, 'it's important to be clear on whose idea it is to start with'. Though 'teachers have to feel solid support' certainly such support cannot be assumed, even from colleagues of longstanding; the messages teachers can expect will be at best 'mixed'. And while there is output of energy, creativity is blunted. Originally a demanding 'process of survival', what then transpires is a situation in which there is 'never enough time to review even half of what's out there and still make things work here in a full-fledged way'. Significantly 'much time is spent in meetings' and in accommodating the someone whose 'always coming through the door'.

Finally, experiment and change are present, but in ways and with consequences teachers find difficult to stay connected with:

> We need time for ourselves, to take a step back and try to get a sense of where we are going, where this energy we're putting out will lead for us, to try to sort through the demands of the program to see which ones are meaningful for us.

Teachers are not curriculum authors in this state of affairs. Yet such authorship is what teachers insist is necessary in commenting about a CIBT program that 'the people who will teach in it have to have a free rein to develop it where their excitement leads', that teachers 'have to develop a program from their perspectives'. Compounding the unlikelihood of this, from another perspective 'it's not clear manufacturers are really aware of what happens in a classroom—when we talk with them at conferences they're not taking notes'. Taken as a whole, it appears teachers' in-class and curriculum autonomy are fundamentally indiscernible in CIBT attempts.

Now it might be countered that to come to such a conclusion is contrary to the tone of the majority of reports available on CIBT programs. Thus, the objection might continue, the outcomes of the evaluation on which this paper rests and the inferences drawn from its data are unrepresentative, i.e., even if the level of teacher autonomy may be minimal in this instance, that does not mean the same applies generally. However, in reply it must be pointed out that unless any existing positive disposition to CIBT has come about in full consideration of such issues as are raised here, then the possibility that teachers are not seen as integral to CIBT curriculum decisions remains distinctly open. That CIBT reportage has not treated such issues in any depth may in fact disguise a bald assumption that it is up to teachers as anonymous variables (little more) to make CIBT work for all concerned.

In consequence, teacher autonomy needs to be addressed much more fully as CIBT efforts increase. As implied earlier, from within education institutions reliance on rhetoric as substitute for action will not suffice when it comes to CIBT; this technology and its curriculum consequences are not fads. It is doubly insufficient under these conditions to ignore, downplay, or not even recognize the now-more-than-ever need to allow teachers the reponsibility to choose curriculum for the students in their classes. So too manufacturers and promoters of CIBT technology must recognize the pedagogic responsibilities and sensibilities teachers have beyond any particular technology-related approach to learning, no matter how attractive that approach might be in isolation. While in-house claims of success may seem less frequent if teachers become integral to all CIBT curriculum processes, long-term success seems precluded without their doing so.

Notes and References

1. As witness to this, note the discussion and recommendations concerning teachers in the slew of regional and national reports of the last decade advocating educational reform.
2. On this see Kurland, D. M. & Kurland, L. C. (1987). Computer applications in education: a historical overview. *Annual Review of Computer Science*, 2.
3. See Gray-Whiteley, P. (June 1990). *Preliminary report of Project TLC: qualitative aspects*. Report submitted to the National Science Foundation Transformed Learning Center Project, Jenks (OK) Public Schools. For these issues elsewhere, see also Gray-Whiteley, P. (Spring 1990) *Barnsdall Schools Special Alternative Instructional Program: qualitative aspects* Report submitted to the U.S. Department of Education Special Alternative Instructional Program, Barnsdall (OK) Public Schools.
4. Recent reports notwithstanding, e.g., (1988) *Power on! new tools for teaching and learning* Washington DC: Office of Technology Assessment.

5. The perspective presented in this section is a composite of the perspectives of five teachers and a director who worked in the project under evaluation; when necessary the project director taught as its substitute teacher. Teacher statements have been thematized and synthesized to provide one 'interview' rather than six; this paper—including the 'interview'—was reviewed prior to submission by five of these six people.

Participants in the study (in addition to the five teachers and director) were students in the program and their parents, a counselor, three central office administrators and three ancillary staff members. Sources of data include observations and interviews conducted with these people during weekly site visits over a period of six months. Observational data were recorded in the form of field notes; interviews were taped in their entirety and transcribed partially.

Paper

Solutions: Just A Matter of Time.... An Effective Staff Development Model for Follow-up and Continuing Assistance in Technology Integration

Kam Matray
Monterey Model Technology Schools Project
Monterey , California

Abstract

California's Monterey Model Technology Schools Project has developed a time and cost-effective model for providing individualized staff development opportunities, follow-up, and continuing assistance for classroom technology integration. This paper reports the rationale, process, and evolution of the ***SuperSub Service*** wherein participating staff are released by assigned "SuperSubstitutes"; these substitutes utilize computer teacher stations and technology-based lessons aligned with district curriculum. Teachers determine the content and pacing of these staff development sessions based on individualized student/ teacher needs. Assistance is provided by project personnel, consultants, but increasingly by in-house trainer-of-trainers in preparation for funding phaseout.

Introduction

In the 1970's—even before the major onslaught of "educational technology integration" and its additional impact on teacher time crunch—local, regional, and national leadership was acknowledging the fact that their teachers were virtually unsupported in time provisions for continuing education and staff development endeavors. Schaefer(1966) aptly described teachers as "functionaries who had been wound up like an old Victrola and were expected to play sweet music forever" with no time to even change the records. It continues to be a widely noted concern that teaching is the only complex vocation whose personnel are not provided with adequate time for inquiry, technical skills acquisition, collaboration, let alone reflection. Many schools, unlike in other professions, expect their teachers to acquire training during off-hours... historically, after a full day of teaching, during evenings, or on Saturdays. The California Postsecondary Education Commission's study on staff development found that "when the salaried workday and work year provide teachers with relatively little out-of-classroom time, teachers' opportunities for *productive* staff development dwindle." (1989)

Enter the '80s and Educational Technology...

The issues of time and staff development are compounded when technology is the focus. According to the Office of Technology Assessment, inservice training in technology has "unique requirements" that distinguish it from traditional inservice activities. Beyond the obvious need for equipped facilities, studies point to the critical importance of follow-up and continuing assistance because staff development must "often overcome the experienced teacher's varying levels of 'technology anxiety.'" Teachers need time to experiment, to collaborate, to commiserate, to practice, to feel confident about using technology in front of their students, many of whom are much more skilled in technology use... time simply to address this complex issue of technology anxiety.

Bruber (1989) aptly addresses the issue of time and technology in her report on the Annual Technology Leadership Conference where her main contention is that educators ought to be viewed as any other professional when it comes to time allotment for learning new technology skills; California Technology Project's Craig Blurton further reflects on time and technology as a factor across the board:

> Technology is, in a sense, another burden on faculty who want to bring it into their courses. If I have 10 weeks of a four-hour class, 40 hours total, and you come to me and say, 'Guess what else you're going to have to do? You're going to model and integrate all the technology skills in that class,' it wouldn't happen. I would simple throw up my hands and quit, and that's what a lot of faculty have done. (Bruder, 1989)

Time and Technology in Monterey, California

The whole notion of providing the "right" time for the "right" kind of support in the "right" way for follow-up and continuing assistance for teachers integrating educational technology into their instruction became paramount during the first two years of implementation of the Monterey Model Technology Schools Project (MMTS), one of California's six "model" technology schools projects. The California Model Technology Schools Program was funded in 1987 by the State of California to establish over a five-year period of time, centers for research, training, and demonstration of the integration of technology in the K-12 educational environment. They are also to provide policy guidance to government and business decisionmakers as to cost-effective implementation of educational technology. (Barkman, 1988)

In Monterey, the 187 teachers participating at the project's four sites (two elementary, one middle, and one high school) developed over 80 project-based Classroom Intervention Plans (CIPs) either individually, in teams, or by grade level. (Matray, 1990) These CIPs:

- plan for integration of technology into an existing curriculum framework area
- articulate with the local school improvement plan
- support one or more of the MTS global objectives
- operationally define the technology and/or the student-centered instructional strategy to be implemented
- identify activities to be implemented by support personnel
- *establish individualized staff development plans*
- describe anticipated products and/or practices for dissemination
- identify CIP development and implementation activities and establish appropriate timelines
- determine development and implementation costs

Initial staff development opportunities for developing these plans reflect a general technology focus during project-wide "technology awareness days" (TADs). The scope of these large group sessions narrows as teachers, now with heightened awareness of available technologies, begin thinking of appropriate applications and instructional integration in specific curricular areas where student needs are most pressing. As teachers complete the planning and development of their CIPs, the need for more individualized follow-up and continuing assistance quickly becomes evident:

> Certain aspects of staff development transcend all potential audiences, while others are quite dependent on the grade level and subject matter orientation of the participants. ...we know that in most situation "one shot" inservices are not very effective, and we know that the staff development needs of secondary school science teachers, for example, are quite a bit different from those of elementary school teachers. (Conery, 1988)

The existing 87 CIPs are in various stages of development and implementation. They range from kindergarten to 12th grade, in all curriculum areas with a mosaic of targeted student groups, student outcomes, and technology emphases. With their attendant, individualized staff development plans, they present a challenge to project personnel in charge of training. Sample CIPs include:

- utilization of selected language arts software in primary whole language instruction
- authoring books for the county Book Affaire using *LegoLogo* and *LogoWriter—Primary*
- home-to-school "Lit-Vid Kits" available in three languages
- multi-media community-based economics projects
- multi-media science instruction
- video portfolios - 21st century assessment tools
- readers' theatre: from computer to camcorder
- writing skills acquisition via international telecommunications
- *"The Electronic Calorie"*, a middle school nutrition course

Some of the Problems

It seems to be a problem of time... MMTS teachers require individualized time for their staff development plans—no two of which are the same. These teachers also need time to implement initial activities outlined in their CIPs—most of which involve planning, previewing materials, as well as collaboration, practicing, and experimenting with new ideas. Some teachers who are more technologically experienced, are at the stage of curriculum development and strategies planning, real time consumers. An initial look at all these staff development needs reveals a degree of commonalty—to a point—which allows clustering of some training activities. But in turn, even these sessions generate requests for presentor-based follow-up and support on a one-on-one basis. Generic staff development for individualized planning and CIP implementation just doesn't seem to work.

Teachers also have begun expressing a desire for support in their own classrooms, with their own target groups and resources. As the CIP process unfolds, it is also becoming obvious that follow-up support and assistance need to address unique teaching styles, management systems, and personalities.

There is a need, in the affective domain, that is increasingly evident and one that plays a large role in the process of teacher functioning as change agent: the need to have some"one" person listen to concerns, fears, stories of total failures, brags of successes, or just plain complaints. The provision of moral support within the historical isolation of classroom instruction during "risky instruction" is a high value item among MMTS participants. Polin (1990) counts this very phenomenon among her four models of teacher support; she also considers this model of support a means for teacher reflection.

During the first year of the project the provision of time for specialized inservice was substitute release. This practice soon generated its own set of problems: 1) too many days out of the classroom... 2) impacted instruction... 3) too often the provision of diluted substitute plans even by the best-intentioned faculty... 4) an unadulterated dislike by teachers of writing substitute plans... 5) overuse of prep periods for training purposes... Toward the end of the semester, teachers began declining full-day substitute release opportunities. When after-school sessions took their place, teacher attendance overall was down.

SOLUTIONS: Just a Matter of Time

The **SuperSub Service** is the result of a search for solutions to these problems. The concept grew out of several brainstorming sessions with project teachers whose ideas are incorporated into the structure of the service. (It should be noted that this early ownership is one of several reasons for the popularity of the service.) The program was so named by excited ten-year-olds who enjoy their sessions with the "SuperSubstitutes" to the extent that they actively encourage their teachers to seek follow-up or continuing assistance!

Relatively simple in concept, it is meeting the needs of teachers at three of the four project sites and addresses many of the problems raised earlier:

1) It provides for weekly release time *during the work day* by a substitute who is assigned to the project for the entire year. During this time teachers can receive individualized, tailored inservice. In this case, Monterey Peninsula Unified School District extended its support of MMTS by assigning an eager and talented early retiree elementary principal to act as SuperSub. This gentleman served for the first two years of the service at the two elementary sites. In September of 1990, the service expanded to include the middle school and now utilizes two SuperSubs, the second of whom is also an early retiree.

2) Teacher-written substitute plans are optional. The SuperSubstitutes who arrive at scheduled classrooms to release teachers are totally "self-contained" with computer teaching station, technology-based lessons appropriate for each grade level, and all necessary materials.

3) Lesson objectives are in alignment with district curriculum and, based on problem solving and critical thinking skill development, address some of the major goals of the MMTS project. Currently, at the elementary level, the service has built its program around Sunburst Communications' *Memory: A First Step in Problem Solving* and *Discrimination, Attributes, and Rules" A Second Step in Problem Solving*. At the middle school, Logo Computer Systems Inc.'s *LogoWriter* is the basis for an independent project-based program supported by the SuperSub Service. (Here, however, the program is closely articulated with math department personnel and is an extension of regular math instruction.)

4) Upon release by the SuperSub, participating teachers meet with project staff, mentor teachers, or trainer-of-trainers for 45 minutes to an hour of individualized assistance. The content of each session is determined by teachers upon sign-up (usually one week previous to each session) and reflects individual CIP staff development plans.

5) The SuperSub Service also provides immediate feedback for participating teachers. At the end of each SuperSub Service day, reports are left with participating teachers which briefly describe the lesson for that session as well as providing a follow-up activity. These reports make reference to pages in the appropriate documentation if further post-computer activities are desired by teachers. (All materials utilized by the SuperSubs are available for checkout at each site.) The SuperSub also includes general comments as to student behaviors if necessary.

6) Each SuperSub maintains site management records to insure continuity in the program. This is another facet that has added to the success of the program; it allows for the sequential delivery of content to participating classes.

7) Participation in the SuperSub Service is voluntary. The session content, scheduled times, and periodicity are determined by the participating teachers and/or their CIP staff development plans. Calendaring and sign-up sheets are provided for the entire year, so teachers can anticipate and request the time for needed individualized inservice, follow-up, assistance, coaching, or planning sessions based on their instructional year. Scheduling of this type allows the staff development coordinator to plan and produce needed materials ahead of time or contact and schedule other trainers such as district mentor teachers.

The structure of this program also facilitates utilization of in-house trainer-of-trainers to provide the requested assistance for their colleagues. With the growing number of talented staff whose own CIPs have made them experts and trainers in their own right, be it in Logo programming, readers theatre, or video editing, many has been the time when the SuperSub releases the classroom teacher and MMTS staff personnel releases the trainer-The SuperSub Service continues to evolve into a program that is highly utilized by project teachers. Service evaluations are made quarterly by participants to increase its effectiveness both on the service side with the SuperSub and on the assistance side with the MMTS staff development personnel. Project statistics show that utilization of the service has doubled from year two to year three; full-day regular substitute use is down 67%; after school sessions [aka *after school burnouts*] are down 43%. Project dollars budgeted for regular substitutes in the past are now being spent by teachers for additional instructional materials or technology conference attendance. These statistics also show an increase in requests for time for:

- demonstration teaching
- collaboration for curriculum "redevelopment"
- observation and coaching sessions
- independent work on expansion of existing CIPs.

The increase use of and requests for trainer-of-trainers parallels the growing pool of in-house experts—possibly a sign that this is one program that can significantly effect the workplace itself as well as deliver the services. This is reminiscent of Mell and Mell's (1990) staff development experiences in Anchorage where a similar "study group model" approach is designated as an alternative means to provide time and motivation from within the organization for staff development. They found that this model met the internal needs of its members and, "when time and money dried up, the team did not." A heartening reference in light of the whims of public funding patterns.

In Monterey's Model Technology Schools' Project addressing the issue of time for teachers and technology has been a joint undertaking. Teachers do need time to experiment, to collaborate, to commiserate, to practice, to feel confident about using technology. They need time for individualized inservice, follow-up, and continuing support during their workday. The **SuperSub Service** is one model that has been successfully implemented to attend these concerns. It is also a process that reflects district commitment and continues to evolve as one solution that is... just a matter of time.

References

Adams, J. (1990). Capacity building, teacher empowerment and accountability: the name of the game for staff development that sticks. *Thrust for Educational Leadership*, 19(5), 13-15.

Barkman, I. (1988). Model technology schools. *California Model Technology Schools*, 1(1), 2-4.

Bruder, I. (1989). New ideas for professional development. *Electronic Learning*, 9(3), 22-28.

California Postsecondary Education Commission. (1989). *Technology and the Future of Education: Directions for Progress*. Commission Report 89-27, Sacramento.

Conery, L., Johnson, V., Moursund, D., and Yates, B. (1988). Effective staff development: preparing teachers to use computer-as-tool [Abstract]. *Proceedings of the 9th Annual National Educational Computing Conference*, 7, 343.

Logo Computer Systems Inc. (1986). *LogoWriter—Secondary*. Montreal, Canada.

Matray, K. (1990). Integrating technology into the classroom: teachers... the important piece in the puzzle. *Proceedings of the 11th Annual National Educational Computing Conference*, 9, 151-155.

Mell, B., & Mell C. (1990). In Anchorage: trials, errors, and successes. In B. Joyce (Ed.), *Changing School Culture Through Staff Development. The ASCD Yearbook* (pp229-242). Alexandria, VA: Association for Supervision and Curriculum Development.

Office of Technology Assessment. (1988). *Power On! New Tools for Teaching and Learning: A Summary*. OTA SET-380. Government Printing Office, Washington, DC.

Polin, L. (1990). Teacher support for technology integration: what worked; what didn't. *Proceedings of the 11th Annual National Educational Computing Conference*, 9, 160-66.

Schaefer, R. (1967). *The School as a Center of Inquiry*. New York: Harper and Row.

Sunburst Communications, Inc. (1988). *Memory: A First Step in Problem Solving*. Pleasantville, New York.

Projects
Multicultural Computer Projects

Computerized Spinoff: A New Look at South Philadelphia High School in the 1990s

C. Tina Petrone, A. Indictor, E. Kelberg, S. Williams
South Philadelphia High School, Pennsylvania

Abstract

The science department has expanded its curriculum to include study in Ecology, Environmental and Health Sciences as preparation for job opportunities. An *Apple Equal Time* grant (1989) launched us into computer applications in the sciences.

Medicinal herb gardens with an ethnic flavor, worm farms for recycling garbage, hydroponics, computer banks of health and environmental career opportunities are some projects at South Philadelphia High School.

Teachers are implementing interdisciplinary instruction by helping students create databases and multimedia presentations using: Apple II GS's, Mac SE's, Mac IIci, scanner, video laser players , HyperCard stacks and on-line data searches. Reports and newsletters are prepared by students using integrated software and an APPLE scanner .

Students, using Apple Link, network with students across the country for comparative health and environmental issues.

Animation Is a Second Language

Joyce Morris and Carol Adamsky
J.H.S. 143, Bronx, NY

Abstract

Our English as a second language (ESL) and bilingual students are improving their English skills and learning scientific concepts by creating original animation on our Apple IIGS computers. In the second year of an Equal Time Grant from the Apple Computer Corporation, children who were walking through rice paddies in Vietnam or working on farms in Haiti a few months ago, are now producing "cartoons", an international language.

Children work in groups of heterogeneous abilities so they can share ideas and improve English skills. Art work is original and drawn using mouse input and graphic tools. Some animated sequences depict predation, ingestion and planetary movement while other presentations focus on environmental issues. Students learn to analyze and express scientific concepts in a creative manner while communicating with their computers.

The icon menu characteristic of graphic programs introduces children to the computer techniques of file saving, retrieval and the sophisticated editing procedures of cut, copy and paste. Children learn how to rotate, size and alter perspective of graphics. They customize brushes and palettes; sequence their screens and compress files for animated effects.

Completed presentations are taped on a VCR and copies made available for students to bring home and share with others.

HyperText Folklife Encyclopedia and a Community-Based K-9 Whole-Life Curriculum

Kathleen Duplantier
St. Tammany School Board, Louisiana

Abstract

In a rural community near New Orleans, students are interviewing, video taping, and writing about the folklife of the area. Folklife includes the traditional face to face culture and customs which the students discover at home and in their neighborhoods. The practitioners and carriers of the traditions, many who are the parents of the students, are the informants at school. This has created a strong community bond to the school. The curriculum at school now incorporates the music, art, and folklore of the cultures that settled this bioregion.

Classroom computers are used to build HyperCard stacks that include recipes and foodways, hunting and fishing lore, family stories, folk crafts and animated folk tales which are also linked to information about the native animals and plants. Curriculum and technology are working together as the students and staff build a HyperText Folklife Encyclopedia. The process of building folk stacks has created visible links in the school curriculum to the life of the community.

Projects
Metacognition

Helping High School Freshman Develop Transferable Computer Skills

Eric K. Neufer
Phoenix Country Day School, Phoenix, AZ

Abstract

Two class periods ("sessions") a week have been set aside for freshmen to receive computer instruction using the school's network of Macintosh computers. The purpose of these ungraded sessions is not merely to impart computer skills, but to impart them in the context of the courses in which the freshmen are currently enrolled. The underlying message which we wish to communicate to the students is that the skills of managing, organizing and structuring information are essential to success in many areas of life.

To provide the students with a direct experience of using the computer as an informational management tool, many of the activities in which students engage during the sessions are actual activities from other courses, in the forms of both homework and supplementary material. The instructor of the sessions works closely with each of the freshman teachers to determine the skills the students need in order to be successful in the courses. This collaboration, in addition to the identification of important basic skills, determines the content of the curriculum.

The presentation will explore a number of issues:

- the content of this year's curriculum
- the challenges of coordinating curriculum across disciplines
- the methods used to encourage students to think about informational management
- the methods used to help students transfer the acquired informational management skills to outside work
- the effect of the sessions being ungraded

Model Technologies School Project

Ronald Rescigno
Hueneme School District, California

Abstract

Abstract unavailable at press time.

Teaching Thinking Skills To All Children

Carol E. Edwards
Southern Coalition For Educational Equity

Abstract

This presentation will describe ways to develop microcomputer based lessons that directly teach critical thinking. It will also describe teaching behaviors that foster skillful thinking. Many attempts for teaching critical thinking meet with limited success because the lessons provide only practice in the exercise of critical thinking. While this may be appropriate for students who already know how to execute thinking skills, it does not help learners who are being introduced to critical thinking for the first time, or novices who have not yet mastered critical reasoning procedures. In this presentation, steps for teaching specific critical thinking skills, including deduction, judging credibility, distinguishing fact from opinion, induction, problem solving, sequencing, classification, detecting bias, etc. will be described.

The assumption underlying this approach to critical thinking is that all children, regardless of their placement on traditional achievement measures can and should learn critical thinking. And, the notion that the relationship between "basic skills" and "higher-order skills" is sequential and hierarchical, that is, that one cannot learn to reason critically until one has mastered rote facts is considered fallacious.

This presentation will provide information about teaching critical thinking within a multicultural context. Specific ways to add cultural relevance to lessons and to reference the environment of learners will be discussed.

Examples of lessons that use science courseware to teach these thinking skills will be presented. The lessons are based on Project MiCRO, a successful program that was created to teach skillful thinking to seventh and eighth grade students using microcomputer based activities to all students.

Special Presentation

The Required Computer Course for Education Majors: 1991 Update

The "Required Computing Course Committee" has been working for several years now. Its mission has 2 major thrusts: 1) to survey a national sampling of teacher education institutions to discover what was included in the computer education course required of all teacher candidates; and 2) to predict national trends in the directions the course was taking. We are currently working on identifying those individuals at each college and university who are most interested in this topic. We then plan to send these individuals a copy of our revised survey and hopefully get a good percentage of responses.

The discussion will cover the following points:

1) development of the questionnaire
2) methodology for conducting and evaluating the survey
3) results of the survey regarding
    a) what is covered in the basic computer education course for teachers
    b) national and regional trends
    c) future implications on computer education for teachers

Panelists:

Joyce S. Friske
Oklahoma State University
Stillwater, OK

Mildred D. Lintner
Eastern Michigan University
Ypsilanti, MI

Charles Mlynarczyk
State University College
Potsdam, NY

Pamela Moore—Chair
Eastern Michigan University
Ypsilanti, MI

Jim Nauta
University of Alabama
Tuscaloosa, AL

Lajeane Thomas
Louisiana Technology University
Ruston, LA

Madge Vining
Columbus, CT

James Wiebe
California State University
Los Angeles, CA

Projects
Curriculum Integration

Transforming Instructional Strategies into Computer Courseware

Annette Lamb
University of Toledo, Ohio

Abstract

Abstract unavailable at press time.

A Computational Tutor for Architectural Design

Patricia Macintosh
Arizona State University

Abstract

Abstract unavailable at press time.

Two to Tango-The School-College Dyad in Promoting Technology

M.S. Vijay Kumar
University of Maine at Machias

William Clark
Superintendent
School Union 102
Machias, Maine

Abstract

School-College partnerships can provide viable and valuable strategies for developing technology use in schools in isolated rural areas.

The University of Maine at Machias and two schools from School Union 102, Machias, have embarked on two seperate collaborative projects designed to prokmote technology in the technology use in rural areas:

1. A multimedia writing project which uses the study of the environment as a vehicle. The project, at the Machias Memorial High School, is suppported by Apple under the Crossroads Educational Grant program. In this project, students working in small research teams will develop a multimedia database of the county's environment and write about it.
2. Project STILE (Supporting Technology Integration in Learning Environments). This project, for training in-service teachers in the use of technology is supported by an IBM grant. The project focuses on training K-8 teachers in the use of microcomputers and multimedia for augmenting the insructional process of Language Arts, Science and Mathematics.

The presentation and discussion in this session will highlight the following:

1. The focal themes underlying the design of both projects, namely: a) collaboration, b) relevance, and c) local environment focus, through a brief multimedia overview.
2. The various dimensions of collaboration involved in all aspects of the project's development and implementation, including teacher training and curriculum development.
3. Problems, strategies and solutions evolving in the ongoing implementation of the projects.

Special Presentation—ISTE
Telecomputing Networks: A Tool to Link Educators

Due to the complexity and unique nature of the teaching profession, it is critical that beginning teachers be provided with structured systematic assistance during their entry year. A successful induction year program will provide critical information, training, technical assistance, and support to beginning teachers as they make the transition to their new school environment. A major reason for attrition from the teaching profession is the lack of a formalized support system. Without support and assistance beginning teachers may become discouraged and abandon their teaching career during the crucial beginning school year. Nationally, it is estimated that 30% of beginning teachers leave during their first two years of teaching and 50% leave during their first four years.

Providing quality supervision and support is a complex task requiring teamwork and cooperation between all components of the education community. Electronic networks can provide the means to nurture and strengthen professional relationships between university faculty, classroom teachers, induction year teachers, interns, student teachers and parents. Some of this support can be delivered by schools and colleges of education via an electronic network linking universities and public schools. This panel will discuss different approaches to linking teachers from geographically diverse regions to form peer-support networks.

Panelists:

Connie Stout
Program Director, Texas Education Agency

Judi Harris
University of Nebraska

Lynne Schrum
University of Oregon

Lynda Haynes
Texas Education Agency

Projects
Telecommunications and Online Databases Across the Curriculum

Information Skills in the K-8 Curriculum

Sandy Lobeck
Yuma School District, Arizona

Abstract

The Library Information Network Curriculum is an outcome of the goal of the Yuma School District No. One library information program. The goal of the program is to facilitate intellectual and physical access to the world of information which learners will experience throughout their lifetimes. In order to address the issues of limited budgets, resources and facilities, we have elected to provide online searching as the key to accessing current resources. Dialog Classmate was selected as our database and is available at our three junior high schools.

Currently, feeder elementary schools may access the database by bringing the students to the junior high library or by formulating the request at the elementary school and sending it to the junior high for student searchers to execute.

Our District participated in one of three Resource Sharing Demonstration Projects funded by the Arizona Department of Education. This project demonstrated resource sharing between the school and public libraries at a local level. We expanded this to include Interlibrary Loan from the Arizona Interlibrary Loan Center. Fax machines, provided by an ECIA Ch. 2 grant, were used for document delivery.

The key to utilization of these resources is the Library Information Network Curriculum which focuses on the partnership between the library media specialist and the classroom teacher. This partnership is facilitated by the team teaching concept in operation in two of our junior high schools, thus expanding the opportunities available for partnerships to develop and flourish.

Using Databases to Teach Research Skills to ESL Students in Freshmen Composition

Beverley Crane
San Jose State Universiyt, California

Abstract

This presentation will describe a project to introduce database searching to English as a Second Language (ESL) students as a tool for teaching research and critical thinking in freshmen composition.

The Electronic Research Center Project

Ann Lathrop
California State University, Long Beach

Abstract

This presentation will describe the new Electronic Research Center developed in conjunction with the Long Beach Public Library, the Long Beach Unified School District, and the Los Angeles County Office of Education to provide access to CD-ROM and electronic databases for teachers and administrators.

Paper
A Case Study for Developing Hypermedia Software

Stacie Hibino
Kathleen Brade

Highly Interactive Computing Environments
The University of Michigan

Abstract

The NASA-Lewis People Tour is an educational hypermedia program to promote a sample of science and math-related careers to junior high and high school students. The software is in the format of a "tour" of the NASA Lewis Research Center where students can "visit" some of the people and projects there. This paper presents the development process of this project as a case study for designing and evaluating educational hypermedia software. Developing such software is a complex process. Existing guidelines can provide a good starting point for design, but articulating strategies or intent in design as well as conducting expert and classroom evaluation are critical.

1. Introduction

As interest increases, the availability of tools for creating hypermedia software also increases, which makes software development faster and more accessible. Unfortunately, these tools are also empowering a greater number of people to quickly develop "bad" as well as "good" programs. Thus, intentional design and formal or informal evaluation are becoming important in the development process.

Literature on hypertext or hypermedia development is being published to address these needs. Some of this literature focuses on guidelines for software development. Although such guidelines can provide a good overview and starting point for design, they tend to be too general. Every software application is different and can be based on a number of factors (e.g., type of application such as simulation or tutorial; content of application such as math, science, general, etc.).

Participating in the full development process (including design and evaluation) can offer a number of insights beyond the scope of existing general guidelines. These insights do not replace general guidelines, but serve to complement and extend them. We found this to be true in the development of the NASA Lewis People Tour.[1] This software is a hypermedia tour of the NASA Lewis Research Center where students can "visit" some of the people and projects. It was developed to promote science and math-related careers to junior high and high school students. This paper presents the development process of this project as a case study for designing and evaluating educational[2] hypermedia software.[3]

The development process that was used can be summarized as follows:

1. Choose a development environment.
2. Evaluate similar software and articulate some specific guidelines.
3. Review existing guidelines.
4. Overall design: select a metaphor and identify components.
5. Practice content collection and prototype software.
6. Collect, reduce, and computerize content.
7. Internal development loop: Implement/Revise, Evaluate, Re-Design.
8. External evaluation loop: Evaluate, Re-Design, Implement/Revise. User testing loop: Test, Evaluate Data, Re-Design, Implement.

The remainder of this introduction will be used to describe our system. The core of the paper focuses on aspects of the development process as outlined above. Samples of what we did, why we did it, and what did or did not work, are described. Special attention is given to the user interface, aspects of design or evaluation which are beyond the scope of more general guidelines found in *HyperCard Stack Design Guidelines*, and our data collection and evaluation process. Finally, the conclusion re-emphasizes the importance of design and evaluation beyond existing guidelines.

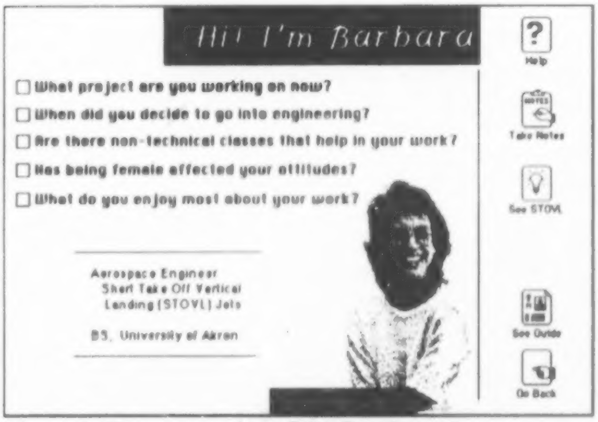
---

[1] This project was sponsored in part by the NASA Space Grant.
[2] The software is mostly promotional, but is "educational" in the sense that it teaches the students about what people in these careers do (what kinds of projects they work on), teaches students not to stereotype the types of people in these careers (by showing them a variety of people in these careers), and teaches them about some of the science behind the projects at a level to which they can relate.
[3] Although this paper will primarily interest developers and researchers, other educators may find it useful for the insights it gives in evaluating hypermedia software (especially the user interface of such software).

1.1 Software Description

Based on accessibility, distribution, and time constraints, *HyperCard*™[*] and the Apple® Macintosh® were selected as the software and hardware development environment. The stack opens with pictures of NASA Lewis and some instructions on how to tour around the center (i.e., how to navigate through the stack). The students are given an on-line tour "guidebook" which is used to visit the center. They can get help and take notes from almost anywhere on their tour.

There are three main kinds of information that the students can visit or view: people, projects, and science. The *people* cards (see Figure 1) allow the student to "interview" some employees at NASA Lewis. Each person interviewed has a *project* card (see Figure 2) about his or her current work. About half of the projects also have information about the science behind them. These *science* cards include more detailed information about how some aspect of the project works.

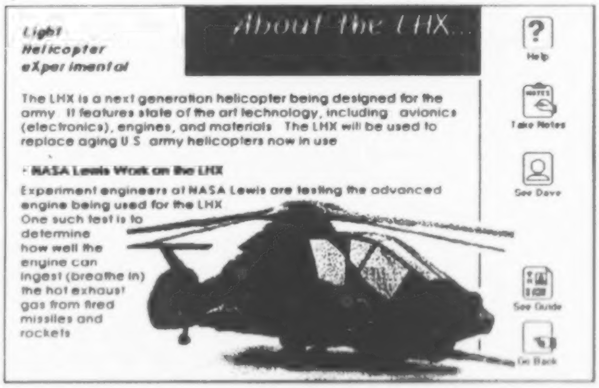

Hi! I'm Barbara
Help
Take Notes
See STOVL
See Guide
Go Back

☐ What project are you working on now?
☐ When did you decide to go into engineering?
☐ Are there non-technical classes that help in your work?
☐ Has being female affected your attitudes?
☐ What do you enjoy most about your work?

Aerospace Engineer
Short Take Off Vertical
Landing (STOVL) Jets

BS, University of Akron

Figure 1. The new version of the ***People*** card.


About the LHX...
Help
Take Notes
See Dave
See Guide
Go Back

Light
Helicopter
eXperimental

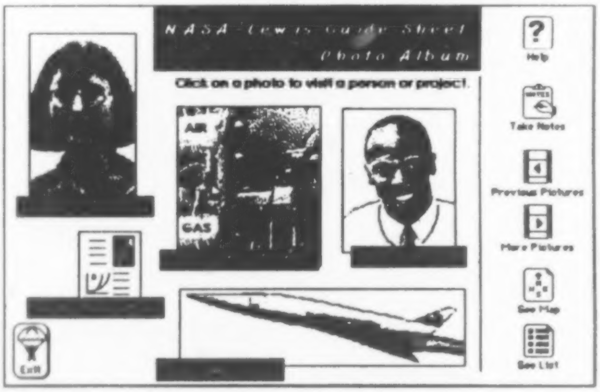
The LHX is a next generation helicopter being designed for the army. It features state of the art technology, including avionics (electronics), engines, and materials. The LHX will be used to replace aging U.S. army helicopters now in use.

• NASA Lewis Work on the LHX

Experiment engineers at NASA Lewis are testing the advanced engine being used for the LHX. One such test is to determine how well the engine can ingest (breathe in) the hot exhaust gas from fired missiles and rockets.

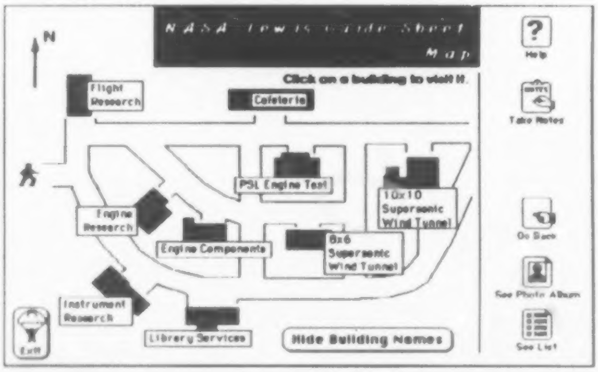
Figure 2. The new version of the ***Project*** card.

The online "guidebook" contains three different indices to information in the stack: a map of some buildings at NASA Lewis, a checklist of all people and projects, and a photo album. In the *checklist*, which is a list with checkboxes, a student can simply click on the name of a person or project to visit that card. Each item in the list is automatically checked after it has been viewed. The *photo album* (see Figure 3) contains a collection of pictures and names of the people and projects. Clicking on a picture on any of the album "pages" takes the student to the corresponding person or project. The *map* guide (see Figure 4) displays the buildings at NASA Lewis where people work. When a student clicks on a building, the building is checked, and an animation shows a person moving from its current position to the building indicated. After the animation, a picture of the building front is displayed. Upon entering, students "meet" one of the people who works in that building. Once students are at a people, project, or science card, they can return to the guide they used last by clicking the "See Guide" button.


NASA Lewis Guide Sheet
Photo Album
AIR
GAS
Exit
Help
Take Notes
Previous Pictures
More Pictures
See Map
See List

Click on a photo to visit a person or project.

Figure 3. The new version of the ***Photo Album***.


NASA Lewis Guide Sheet
Map
N
Flight Research
Cafeteria
PSL Engine Test
10x10 Supersonic Wind Tunnel
Engine Research
Engine Components
8x6 Supersonic Wind Tunnel
Instrument Research
Library Services
Hide Building Names
Exit
Help
Take Notes
Go Back
See Photo Album
See List

Click on a building to visit it.

Figure 4. The new version of the ***Map*** index.
Notice the one-line instruction below the card title.

---

[*] In HyperCard, the basic unit is a "card" composed of text fields, buttons, and bitmap graphics. A card is essentially a screen, and the set of cards in a file or program is known as a "stack."

BEST COPY AVAILABLE

2. Evaluating Similar Software

The Apple guidelines suggest that reviewing other stacks before design can be helpful for generating ideas and seeing examples of how others approach a task similar to yours. We reviewed five programs and found this to be true. In addition, we found that evaluating other software helped us to gain a better understanding of "bad" and "good" stack design. Some examples of bad design can be associated with feeling lost or frustrated, being unable to access information, or being confused by the screen (not knowing what to do next). Some aspects of good design can be indicated if you find the software easy to use (not confusing), informative (meeting goals), or being able to find the information you are looking for (not getting lost). In trying to identify the sources of bad and good design, we were practicing the articulation of purpose and intent for design.

3. Using Existing Guidelines

In general, some of the guidelines included in the Apple reference seem like common sense, but one might be surprised to see how often they are violated. We found that evaluating other stacks and re-discovering some good and bad aspects of design helped us to understand the guidelines in a more practical manner. The guidelines in the reference supported and solidified many of our findings, and vice versa. Guidelines can be used as a checklist for evaluation as well as a starting point for design.

4. Strategies for Our Overall Design

After reviewing stacks and obtaining some guidelines, we were ready to begin the overall design of our stack. This overall design was separated into four parts: specifying a focus which would be a "solution" to the task at hand, choosing a metaphor, determining navigation methods, and planning for user testing.

Specifying a focus. Recall that our main goal was to interest students in math and science-related careers. Research by Hill, Pettus, and Hedin (1990) indicates that one of the main factors effecting science-related career decisions appears to be personal contact with a scientist. That is, students may consider a science-related career if they have a role model—if they know a scientist. Thus, we decided that a reasonable approach for our software would be to include real interviews with professional role models. We would gather this information by conducting actual interviews with people who had jobs in our focus careers.

Choosing a metaphor. We wanted to present the role models and their projects in the most natural way as possible, so that the students might get a sense of what it might be like to actually visit some people at NASA Lewis. Since the Lewis research center is so large, being spread out across a number of buildings, we felt that a "tour" and "interview" metaphor would be a natural one to use. These metaphors, when implemented, would allow the students to "tour" NASA Lewis and interview people along the way.

Determining navigation methods. In an introduction and survey to hypertext, Conklin (1987) describes a "disorientation problem" that is endemic to hypertext systems. This problem, also known as getting lost in "hyperspace," has to do with the complexity formed by providing a number of paths for the user to follow. That is, as the network of the information and paths representing the software gets larger, the users may find it more difficult to understand where they are in the system and how they might get to where they want to be. Although we were working with hypermedia and not hypertext, the same problem could be encountered. Two aspects of the software addressed this issue—a relatively simple stack structure, and a mechanism for indicating where the students had been. The stack structure was relatively simple in that all information was accessible within a click or two of the indices, and a button was provided for the user to get back to the guide index from every card. The checklist and the map both provided check marks to indicate what the student had seen. No matter how a student got to a particular people or project card, the check boxes on the checklist indicated if the corresponding information had been visited.

Because we expected users to have different working styles, we felt it was important to provide different ways in which the students could access the information. Offering three navigation indices allowed the student to use a preferred method of navigation—if one mode was too boring or too slow maybe another one was more interesting or quicker. At the same time, we kept in mind that the lower the mental effort required to attend to navigation (or the operation of the software), the higher the attention the students could give to its content (Kozma and Bangert-Drowns, 1987). This was addressed by matching the different indices provided to the overall "tour" metaphor. We did this by providing a tour guidebook which tied the three navigation modes together as described in the section on Software Description.

Planning for user testing. We knew from the start that we would eventually be testing our software with real users and that data would have to be collected during this testing. The data would need to be detailed to provide a clear understanding of how our stack was being used. For these reasons, we developed a logging mechanism within our software to unobtrusively trace student interactions (i.e., mouse clicks). The resulting trace file was automatically saved when the student exited the software. Another tool was developed to automatically play back the trace files. This play back allowed the developers to get a quick overview of any interface problems the students encountered. It was important to think about data collection before implementation, since it would have been very difficult to add it later.

5. Practicing Content Collection and Prototyping

Once we had an overall design in place, we could begin practicing content collection and prototyping. In our case, collecting content was not a trivial process, so practice was essential. Interview questions were derived based on the information we would convey to the students (e.g., information about their work, about why they decided to work at NASA, etc.), various equipment were collected, and then practice interviews were conducted with some professors on campus. The interviews were tape recorded, videotaped, and the interviewees were photographed. We decided to collect information in different media formats in order to have more than one available, and so that the different formats would be available for an expanded version of the software. Practicing the interviews gave us feedback on: the interview questions, the amount of time it would take to set up the equipment, how long to let the interviewee speak, and what types of pictures and sounds gave the best results.

We developed a prototype of our software using the practice data collected. Due to the visual nature of our "tour" format, our software would include a number of pictures. In addition, we were hoping to include some sound, even if that meant short sound effects only. Since both pictures and sound could be limiting factors to the size of our stack, the prototype gave us a mechanism for estimating approximately how many interviews we might be able to include.

6. Collecting, Reducing, and Computerizing Content

We interviewed eighteen people with various jobs and backgrounds at NASA Lewis. In order to "computerize" the media collected, the interviews had to be transcribed, reduced, and entered as text into the computer, and the pictures needed to be digitized. The text of the interviews were transcribed and reduced to approximately five questions and answers. In order to generate more interest for the students, these five questions were different for each of the people interviewed, but all had a common thematical base. For example, one theme was on projects: every person had a question about what project they were working on at NASA Lewis. Some of the other main themes included: schooling, reasons for career choice, and affective attitudes about work. The various types of video images collected were selected in many phases and finally reduced to the pictures we needed to go with the various pieces of text.

7. Internal Development

The software was implemented using the prototype as a base and began in parallel with content preparation. With a development team of six, we could cycle through the development loop internally—implement, evaluate/revise, re-design. That is, part of the group could focus on design, part could focus on content, and part could focus on implementation. One part of the team could then critique another. In addition, we also received some feedback from some individual students at this point, as they became available and our software became more robust.

8. External Evaluation

Once all of the content was in place and basic functionality was obtained, the evaluation stage shifted from internal to external focus. The external evaluators can be classified into two categories—experts and model users.

8.1 External Evaluation by Experts

The Apple guidelines describe the importance of evaluation, but not necessarily evaluation by external experts. Experience, guidelines, and good intuition can lead to good design, but even experts can make mistakes. The longer we worked on the project, the more we became biased towards the design choices we had made. Evaluation by experts external to the development team could help us to identify possible problems or inconsistencies in our design.

Experts from three areas were chosen corresponding to the three overall aspects of the software: content, graphic design, and user interface. The content could be evaluated on two levels—accuracy and use of language. The accuracy of our content was externally evaluated by project advisors and people at NASA Lewis. Evaluating the use of language would involve checking to see if the text was written at an appropriate reading level for the targeted audience. Arrangements for this evaluation are currently in progress.

Two experts in user interface design and one in graphic design evaluated our software in individual sessions with one of the developers. During these sessions, some possible problematic areas were identified. Some of these were minor, while others were larger. Problems that we had overlooked were immediately fixed; those related to preference or which seemed to go against our design strategies were noted, in order to compare with the results of user testing. For example, one of the interface experts thought that the checklist did not add much information—that the names of the people and projects did not hold much meaning out of context (i.e., out of the context of pictures, a context which was provided in the photo album). She felt that we should consider removing it altogether and replacing it with the photo album. Since we had strategies behind including each of the indices, we were hesitant to do this. We decided to wait and see what the users thought about the checklist before making such a major modification.

8.2 External Evaluation by Model Users

User testing is important, because we were not the target users, and we could not predict what they would want or understand. The focus of the classroom-like testing described below was on the user interface. We were interested in seeing if the students understood our tour and interview metaphors, and in identifying any areas where they had problems. We were also interested in hearing what they had to say about the three indices—if they liked having a choice, and which one they preferred the best.

A classroom of 29 students tested the stack. The students were tenth and eleventh graders (mainly 15 and 16 year olds) who were primarily minority students from various cities in

Michigan. They were visiting the University of Michigan for a summer camp with computer classroom time as part of their daily schedule. The number of male and female students was almost evenly split. Greater than 80% of the students had some computer experience, though only a smaller number of them had used the Macintosh before coming to camp.

The testing took place in a computer lab on the university campus and most of the students tested the stack alone, although some explored in pairs. We allowed testing in pairs since we thought that in a normal classroom setting, it would probably not be unusual for students to share a computer. The students were told that the purpose of using the software was not only for them to find out about NASA Lewis and some of the people who work there, but that they were also helping us to evaluate the design of the stack.

Four methods were used for collecting data: questionnaires, observations, trace files, and informal debriefing. The questionnaires collected some background information about the students (e.g., grade in school, computer experience, etc.) which they filled out before they used the stack. The rest of the questionnaire was used to collect student comments evaluating the stack. This part of the questionnaire could be filled out during and/or after the student used the stack. It included questions about student preferences (e.g., which person they liked the best, what they thought about each of the indices) and whether there were any confusing parts of the stack. The observational data were taken by four of the developers who observed the students as they used the stack. The trace files collected all button and field clicks that the students made. When testing was completed, the students were moved to another room for informal debriefing where the four developers asked the students some additional questions (e.g., which index did you like the best?).

The data collected from the classroom testing provided a number of insights related to the use of the three indices. First of all, both observations and the play back of trace files indicated that a number of students did not read the explanations about how to use each of the indices, that they were reluctant to use help, and that it was not obvious to all of them that they could click on buildings on the map or that they could click on pictures in the photo album. We decided to resolve this issue by including a one-line instruction on each of the indices, indicating what was clickable. If a student clicked on the help button, then more detailed information would be given.

As we predicted, students liked having more than one navigation index and being able to use the one which they preferred. In addition, they liked being able to easily switch between the indices. A show of hands during the debriefing session indicated that about half of the students liked the map the best, and that the rest were split fairly evenly between the checklist and the photo album. In addition, the questionnaires supported this. One student wrote that the checklist was "Cool (saved time—very brief)" while another student wrote that it was "interesting, but boring." A couple of comments about the photo album were that "it was a good way to find places" and that "it added a lot to the program." And finally, the map was thought to be "good, it showed how to get around and where the people worked" as well as "a good way to show the whole area."

The map seemed to successfully reinforce the tour metaphor since during the debriefing, the students made comments that the stack was like being able to visit NASA-Lewis without going there and that we should make similar stacks for the Kennedy Research Center, other companies, or even a whole city.

9. Conclusion

Developing software is a complex process. In developing *educational* software, one must think about the content, structure, *and* interface to the program. A good interface will reduce the cognitive load required to operate the software. A poorly designed interface will reduce the mental effort available to attend to the task and might also cause interference in the student's understanding (Frye and Soloway, 1987). In developing *hypermedia* software, additional consideration must be given to navigation. Guidelines provide a good starting point for design, but articulating strategies or intent in design as well as conducting formal or informal evaluation are critical parts of the development process. When you are building a program that lots of people will use, testing and evaluation become essential, since it is difficult to predict a user's behavior.

Acknowledgments

The rest of the NASA-Lewis People Tour development team included: Scott Carter, Jill Kliger, Amy Polk, and Kevin Quarderer. Fred Zinn reviewed the graphic design of the stack and provided other helpful comments and discussions related to stack design and redesign. Special thanks to Professor Elliot Soloway for his advice and support of the project, and review and comments on this paper; to Professor Joseph Eisley for his advice and support in the project; and to Mark Guzdial for review and comments on both the software and this paper.

References

Apple Computer, Inc. (1989). *HyperCard Stack Design Guidelines*. New York: Addison-Wesley Publishing Company, Inc..

Conklin, J. (1987, September). Hypertext: An Introduction and Survey. *IEEE Computer*, 17-41.

Frye, D., Soloway, E. (1987). Interface Design: A Neglected Issue in Educational Software. *CHI'87 Conference Proceedings*, 93-97.

Hill, O. W., Pettus, W. C., Hedin, B. A. (1990). Three Studies of Factors Affecting the Attitudes of Blacks and Females Toward the Pursuit of Science and Science-Related Careers. *Journal of Research in Science Teaching*, 27(4), 289-314.

Kozma, R., Bangert-Drowns, R. (1987). *Design in Context: A Conceptual Framework for the Study of Computer Software in Higher Education*. Ann Arbor: Board of the Regents of the University of Michigan.

Paper

Teaching Business Policy Using Multimedia Game Simulation Support

Richard Cobb
Jacksonville State University
Jacksonville, Alabama

Abstract

This paper reports on the efforts of both faculty and staff at one university business school to integrate successfully computer simulation, along with other support tools, into the Business Policy and Strategy course. This was done using a multimedia approach including mainframe computer software selection, the development of interactive decision support software, and instructional video production. Our efforts were aimed at overcoming traditional barriers to convenient computer use and at using available media resources to enhance the value of the computer as a teaching and learning tool.

Introduction

The Business Policy and Strategy course, often referred to as 'The Policy Course', is considered to be a capstone requirement for majors of most university business schools. While course content and degree of subject emphasis may very from school to school, most policy courses will include basic theory used in combination with teaching and learning tools such as cases, outside-of-class exercises, and business simulation games. The ability to use these tools and to gain greater benefits from them can be enhanced through multimedia applications. These applications may include the use and adaption of computers, video, or other media resources designed to help overcome barriers to understanding and learning. This paper will review the work done by both faculty and staff to integrate successfully computer simulation and other media tools into the Business Policy course at one university business school. Here, degree of success was determined by the ease with which students learned to use the computer and other related media. Special emphasis was placed on reducing the amount of time needed to make simulation decisions. This would increase the amount of time available for understanding and using simulation output.

Use of Computer Simulation

Many papers have been written about the importance of the Business Policy course and the need to integrate simulation as a learning tool supporting course theory requirements. An example emphasizing this importance is the work done by Smith in which he reported that simulation, used as a method for learning, provides a unique opportunity for students to practice making decisions in a management environment [5]. Boyd, Kopp, and Shufeldt also reported on other studies that support the use of simulation games as an important method for learning [1]. In a related work, McCartney, Richardson, and Callarman reported that computers and business simulation games have been used for many years in teaching Business Policy [3]. These and other findings provide additional information on the use of computers as teaching aids or report on the widespread support for the use of simulation as a problem solving, teaching, or learning tool. This support for the use of simulation techniques and the development of new simulation applications can be closely matched to the pace of computer development [4].

The fundamentals supporting simulation games all tend to conclude that the practice of management, even in a simulated environment, takes on new meaning when students are allowed to make decisions and test various 'what if' scenarios applied to a hypothetical business environment. This is made even more important today as we see real world business problems growing in both scope and complexity. Business school students can therefore benefit by using this valuable learning tool. The skills needed to compete successfully in a simulated industry are complemented by the theory presented in the course. This helps students to understand the interdependent nature of the various functional business areas and also to teach the value of teamwork and cooperation. A very competitive environment usually develops in each class. This helps to increase the level of motivation in the course.

Simulation Game—Multimedia Support

Although this paper does not focus on the criteria for simulation software selection, it seems appropriate to add here that any simulation game must be compatible with the needs of the course and the available computer facilities. With these as basic selection criteria, the simulation game chosen for use in our Business Policy course was *The Business Management Laboratory (BML)* by Ronald Jensen and David Cherrington [2].

This is a widely used business simulation software package and well suited to the Business Policy course at our school. In this business simulation game, student teams follow a familiar theme in which they are organized into competing companies in a defined industry. It is necessary that they plan and make decisions that represent quarters of simulated business activity for their firms. To aid this activity, multimedia support has been developed and provided to each team. These multimedia applications were judged to be successful if they reduced the time needed to learn to use the simulation game and enhanced its value as an aid to learning. The following discussion will further explain these media support applications.

Data Entry Interface. The BML software runs on an IBM 9370 mainframe computer using the Multi-User-System-for-Interactive-Computing (MUSIC) operating system. The data entry interface was designed for this system in order to help support different teachers involved in teaching multiple sections of this course and to aid students with different backgrounds and majors that would use the mainframe system. The data entry interface software applied to this media helped to meet these objectives. This software was developed by the staff at our school's Academic Computer Services division. With BML, team decisions represent one simulated quarter of business activity. In the Business Policy course, thisusually means that one decision set per team is entered into the computer each week. The data entry interface software is interactive and allows students with little mainframe computer experience to enter decision data quickly and accurately. This is done by using menu driven subroutines.

Initially, students will sign-on with their unique team ID and password. This provides security for their decisions and allows a convenient method for the game administrator to retrieve and run their decisions. The first set of screen options presented to the teams is shown in figure 1 and represents the BML main menu. In this menu, four options are available and are selected as needed by the student teams.

By selecting option1, thestudent teamscan receive additional information which can be helpful to the noviceuser of the computer or the data entry program. By reviewing this information, students can answer many of the questions which would normally require additional assistance from an instructor.

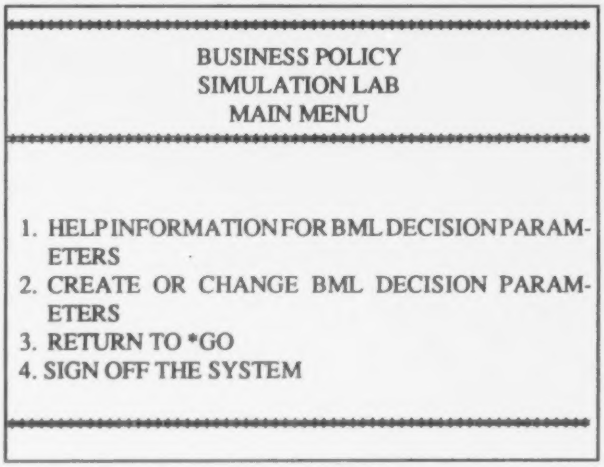
The primary use of this software is defined by option 2. This option will take the teams into the interactive data entry phase and is completed in two successive screen sessions as shown in figure 2 and figure 3. At each data entry category, the students need only to use the tab key to advance to each successive data decision category. They will then key in any new or changed data before they tab to the next category. After the first of the two data entry screens is reviewed, they press the enter key to move to screen two. A comprehensive check for syntax mistakes is made in this data entry phase, and the students will by prompted by an error message if such a mistake is detected. The data entry session will automatically return to screen 1 and the program menu.

Option 3 will automatically return terminal operation to the *GO mode. This mode will allow the student to use the mainframe for other functions.

Option 4 is an automatic sign-off command. This was included to provide student teams with a convenient method for ending terminal sessions.

After all team data are entered, the simulation program administrator will run the program. The results of team decisions can then be evaluated, and strategy for the next quarter planned. This process is common to all sections of the course. By using the interactive software, individual expertise relative to the mainframe system and the simulation program can be reduced for both teachers and students.

Instructional Video. The interactive data entry program is further supported by an instructional video produced by the JSU Management Department. This video gives the student teams an easy to use source of basic information on the simulation game and the data entry process. The instructional video is a simple step-by-step presentation showing how to use the mainframe computer terminal to enter simulation decision data. The tape runs for approximately fifteen minutes and takes the student teams through the basic theory of what decisions to make, how to make these decisions, and how to enter these decisions in their computer files using the interactive data interface. Class time that would normally be used explaining this procedure can thus be reduced, and emphasis can be shifted to the theories of strategic planning.

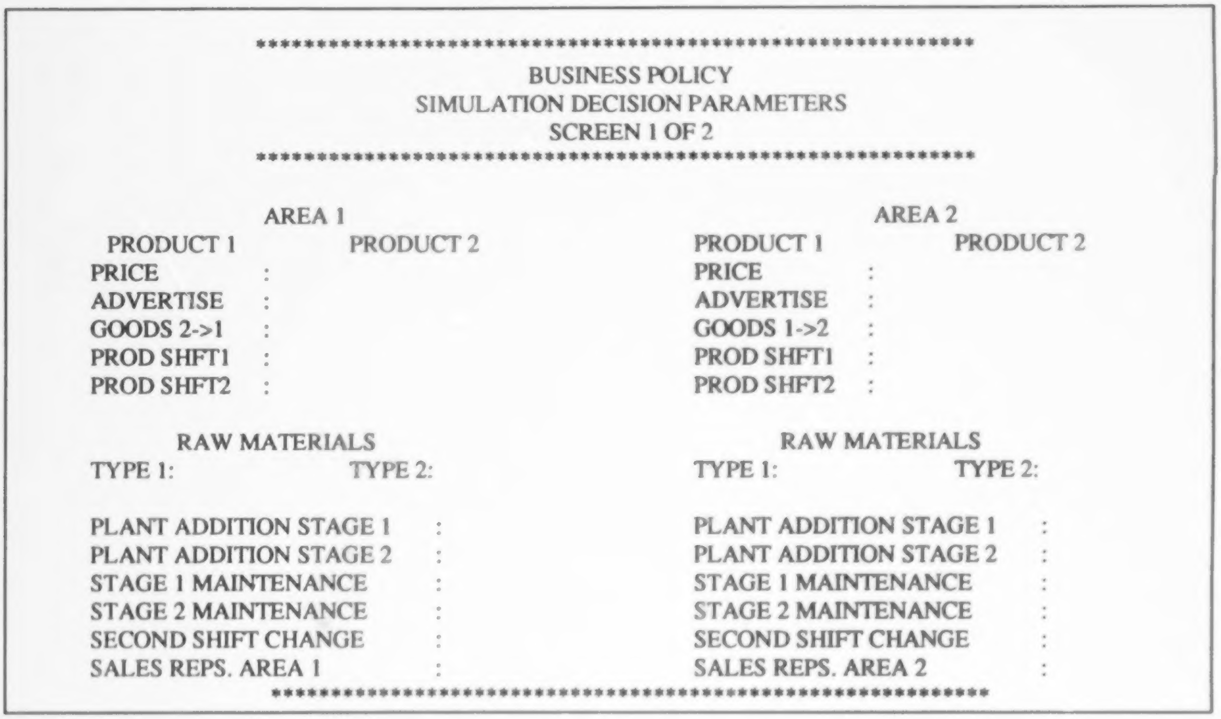


```
***********************************************************
                    BUSINESS POLICY
                    SIMULATION LAB
                      MAIN MENU
***********************************************************

 1. HELP INFORMATION FOR BML DECISION PARAM-
    ETERS
 2. CREATE OR CHANGE BML DECISION PARAM-
    ETERS
 3. RETURN TO *GO
 4. SIGN OFF THE SYSTEM

***********************************************************
```

Figure 1. Simulation Laboratory Main Menu



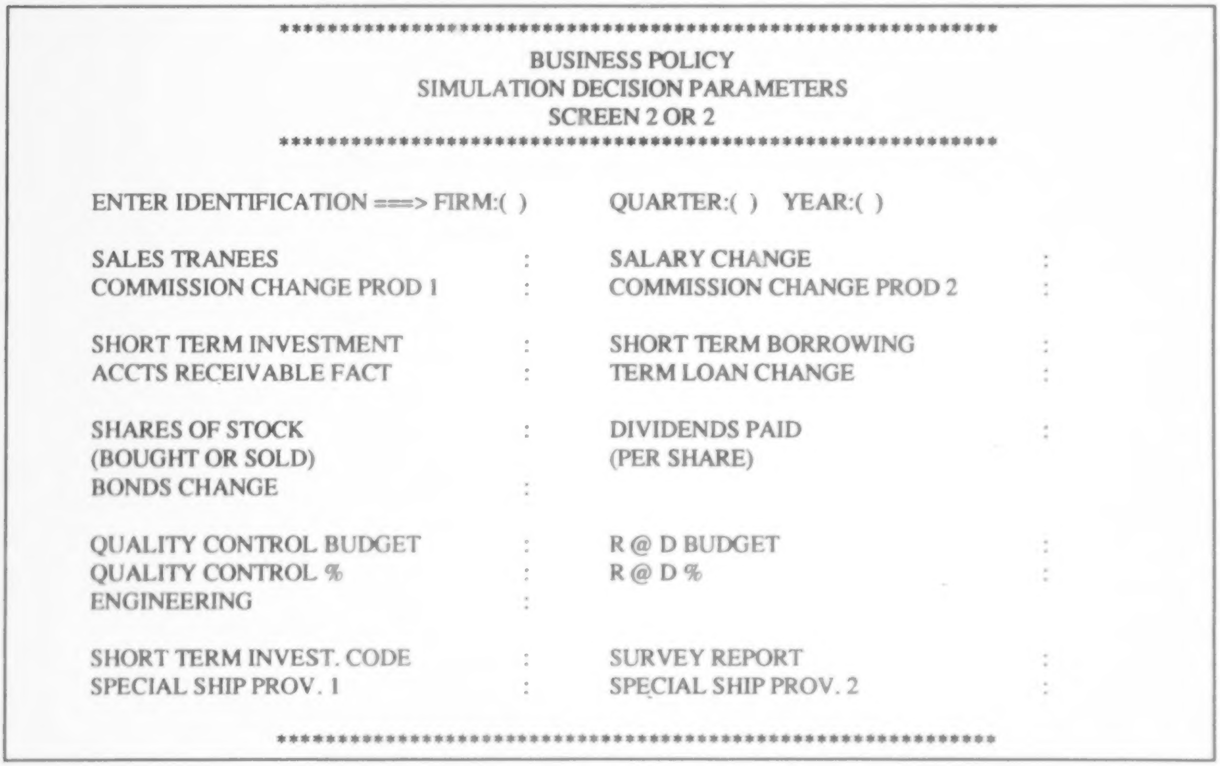
```
               ****************************************************************
                                  BUSINESS POLICY
                           SIMULATION DECISION PARAMETERS
                                   SCREEN 1 OF 2
               ****************************************************************

            AREA 1                                              AREA 2
  PRODUCT 1             PRODUCT 2                    PRODUCT 1              PRODUCT 2
  PRICE          :                                   PRICE          :
  ADVERTISE      :                                   ADVERTISE      :
  GOODS 2->1     :                                   GOODS 1->2     :
  PROD SHFT1     :                                   PROD SHFT1     :
  PROD SHFT2     :                                   PROD SHFT2     :

         RAW MATERIALS                                      RAW MATERIALS
  TYPE 1:               TYPE 2:                      TYPE 1:                TYPE 2:

  PLANT ADDITION STAGE 1      :                      PLANT ADDITION STAGE 1      :
  PLANT ADDITION STAGE 2      :                      PLANT ADDITION STAGE 2      :
  STAGE 1 MAINTENANCE         :                      STAGE 1 MAINTENANCE         :
  STAGE 2 MAINTENANCE         :                      STAGE 2 MAINTENANCE         :
  SECOND SHIFT CHANGE         :                      SECOND SHIFT CHANGE         :
  SALES REPS. AREA 1          :                      SALES REPS. AREA 2          :
                 ****************************************************************
```

Figure 2. Simulation Decision Parameters for Screen 1



```
                 ****************************************************************
                                   BUSINESS POLICY
                            SIMULATION DECISION PARAMETERS
                                    SCREEN 2 OR 2
                 ****************************************************************

  ENTER IDENTIFICATION ===> FIRM:( )       QUARTER:( )   YEAR:( )

  SALES TRANEES                      :     SALARY CHANGE                      :
  COMMISSION CHANGE PROD 1           :     COMMISSION CHANGE PROD 2           :

  SHORT TERM INVESTMENT              :     SHORT TERM BORROWING               :
  ACCTS RECEIVABLE FACT              :     TERM LOAN CHANGE                   :

  SHARES OF STOCK                    :     DIVIDENDS PAID                     :
  (BOUGHT OR SOLD)                         (PER SHARE)
  BONDS CHANGE                       :

  QUALITY CONTROL BUDGET             :     R @ D BUDGET                       :
  QUALITY CONTROL %                  :     R @ D %                            :
  ENGINEERING                        :

  SHORT TERM INVEST. CODE            :     SURVEY REPORT                      :
  SPECIAL SHIP PROV. 1               :     SPECIAL SHIP PROV. 2               :

                 ****************************************************************
```

Figure 3. Simulation Decision Parameters for Screen 2

Conclusions

The efforts described in this paper were aimed at helping to bridge the gap between the student and the successful use of a computer simulation game in the Business Policy course. This involved overcoming traditional barriers to computer use and has helped to make the computer more useful as a teaching and learning tool. The development of the interactive data entry software and the instructional video production has helped to meet these goals. Also, this has helped to integrate and further strengthen the important role played by simulation games as a learning tool supporting the Business Policy course.

These multimedia efforts have enhanced the value of Business Policy as a capstone course in which students learn to apply their knowledge of different functional business areas to the decision making environment.

References

[1] Boyd, C., Kopp,D., & Shufeldt, L. (1984). Effects of Academic Specialization, Highest Earned Degree, Class Size, and Geographic Location on Grade Determination Factors in the Business Policy Course. *Proc. of the Southern Management Assoc.*, 225-228.

[2] Jensen, R., & Cherrington, DJ. (1984). *The Business Management Laboratory*. Homewood, Illinois: BPI/Irwin.

[3] McCartney, W. W., Richardson, W.D., & Callarman, W.G. (1987). The Use of Computer Based Decision Support Tools in the Business Policy Course. *Proceedings of the Southern Management Association*, 340-342.

[4] McLeod, J. (1986). Computer Modeling and Simulation: The Changing Challenge. *Simulation, 50*(3), 114.

[5] Smith, J. R.(1988). Using Business Simulation in the College Classroom. *Proceedings of the Southern Management Association*, 105-106.

Projects
Integrating Laser Disks in Your Classroom

Interdisciplinary Instruction Using Laser Videodisc

Jim O'Brien
Metropolitan School District of Perry Township
Indiana

Abstract

Teachers at Perry Meridian High School within the MSD of Perry Township have used laser videodisc to create interdisciplinary instructional units. The project consists of the work of three interdisciplinary teams. One team, comprised of biology, algebra, and English teachers developed a genetics unit using the laser disc application Principals of Biology. A second team of government, mass media, and salesmanship teachers used the Vote '88 laser disc application to develop an election unit. U.S. History and ethnic literature teachers comprised the third team and use the Martin Luther King Jr. laser disc application to develop a civil rights unit.

Project evaluation will be conducted in three areas. 1) Student pretest/posttest achievement data between control and sample groups will be analyzed to determine if use of laser video disc significantly affects student achievement. In addition, achievement data from students identified as visual learners will be compared with similar data from students identified as visual learners will be compared with similar data of other sample students for significant differences. 2) Participating students and teachers will complete an attitude survey to determine if the project fostered differences in student motivation and changes in participating teachers' instructional methods. 3) Spinoff activities from the project will be documented and analyzed.

Portable Multimedia Unit Project Report

Gerald Williams
Inland Area CUE, California

Abstract

Last year at Lucerne Valley Elementary school we put together a portable multimedia cart we originally called a Portable Science Unit. Since that time however, this system has also been used extensively for Language Arts and Social Studies.

The cart consists of: 1. A 26" color monitor, 2. A Macintosh Classic computer, 3. A Pioneer V2200 Laser Disc Player, 4. A VCR, 5. A high resolution mini color camera with an adapter to allow projection from a microscope, 6. A quality microscope, 7. An ImageWriter Printer, 8. A modem, and 9. A LCD Projection unit. All of these items are on a cart that allows the unit to be easily transported from room to room.

The use of the high resolution color camera placed over the eyepiece of a microscope allows for whole class viewing of slides. The camera is also used to view teacher demonstrations or printed materials. Whatever takes place under the camera can be recorded and then edited on the VCR for students who were either absent or who need review.

We use Opinions Data Windows on Science Laser Disc series for our science program, National Geographic's GTV for Social Studies, and laser discs that go along with our core literature to develop such concepts as plot, character and theme as well as to stimulate creative writing.

Repurposing Video and Audio Discs for Interactive Classroom Use

James Wall
University of South Alabama

Abstract

Abstract unavailable at press time.

Author Index

Abate, Ronald J. 76
Achey-Cutts, Patty 224
Adams, Charles F. 66
Adamsky, Carol 329
Adamson, Eadie 171
Adickes, Martin 114
Aiken, Robert M. 132
Ambrosino, Jodie 2
Anderson-Inman, Lynne 4
Anton, Jim 244
Appert, Christine 106
Ashton, A. 281
Auck, Chuck 65
Bankhead, Elizabeth 260
Basham, David 78
Bauder, Deborah 285
Beaver, John F. 160
Becnel, Shirley 122
Beekman, George 78
Bell, Laura 262
Bell, Sharon 48
Bessman, Marcelle 203
Beytin, G. 244
Bhote, Nina P. 132
Biglan, Barbara 288
Bjelland, Kathy 284
Blurton, Craig 156
Bombara, Stephen F. 110
Boudrot, Thomas E. 1
Boyer, Barbara A. 26
Brade, Kathleen 336
Branscomb, H. Eric 33
Branzburg, Jeffrey 260
Braswell, Ray 258, 316
Braun, Lud 210
Brett, George 251
Brody, Joyce 80
Brown, Gary 175
Brownell, Gregg 155, 203
Brownell, Nancy 155
Brumbaugh, Kenneth E. 140
Calderon, Enrique 211
Campbell, Tom 206
Cardinale, Loretta 25
Carey, Doris 276
Carey, Regan 276
Castle, Shari 201
Cathcart, Gloria 263
Cathcart, W. George 188
Chaille', Christine 288
Chandler, Terrell N. 288
Christy, Jerry 282
Clark, Chris 108
Clark, William 333
Clément, D. 226
Coats, Kaye 24
Cobb, Richard 341
Cochran, Paula 316
Collis, Betty 195
Conger, Cathy 226
Conway, Catherine F. 63
Coon, Lance 24
Cordes, Cathy 5
Court, Claudine 211
Cowan, Hilary 125
Crane, Beverley 335
Croft, Steven 33
D'Ignazio, Fred 48, 319
Dana, Ann 246
Daughenbaugh, Richard 287
Davidson, Hall 89
Davis, Lora Kawka 289
Deffis, Gustavo R. 211
Derycke, A. 226
Diaz-Rico, Lynne 278
Dill, Nancy 242
Djang, Philipp A. 164
Dodl, Norman R. 97
Doose, Lyn 33
Doss, David L. 80
Doucette, Debora 282
Dubitsky, Barbara 127
Dunnock, Joanne M. 28
Duplantier, Kathleen 329
Eber, Judith 204
Edens, Helen 315
Edwards, Carol E. 330
Ehinger, June 24
Elliot, Alison 25
Ellis, James D. 171
Elowitz, L. 244
Elowitz, M. 244
Elting, Susan 141
Espinoza, Sue 246
Flanagan-Hyde, Peter 63
Ford, Mary Jane 209
Foreman, Deidre 31
Fowler, David 2
Frank, Diane 116
Fredman, Alice 209
Freedman, Debra 262
Friedman, Robert 242
Friesen, Chuck 241
Friesen, Patricia 226
Friske, Joyce 284
Fronk, Robert 125
Frye, Douglas 2
Gale, Douglas 277
Gershner, Vera T. 290
Goldstein, Gideon 317
Goltz, Carol 116
Golub, Jeffrey 317
Gonser, Connie 158, 175
Gonzalez, Rocio 211
Goodman, Carol 171
Goulding, Phillip G. 171
Grady, Ann 247
Graeber, Janet 24
Gray-Whiteley, Peter 320
Greenberg, Richard 33
Greiner, Jill 283
Gritsouk, Anna 204
Gross, Naomi 106
Gunn, Cathy L. 40
Gunn, Tronie 92
Guzdial, Mark 16
Hamilton, Julie 201
Hancock, Chris 114
Hancock, Martha 315
Hancock, Vicki 242
Handler, Marianne 246
Hanor, Joan 108
Hare, Karen 33
Harris, Judi 30, 285
Hart, Pat 5
Haverfield, Mary S. 258
Healey, Deborah 214
Healy Stearns, Peggy 31
Heller, Rachelle S. 203
Henderson, Sheri 153
Hernandez, John 287
Hess, Robert 283
Hibino, Stacie 336
Hilgenfeld, Robert 74
Holiday, Sandee 224
Horn, Sue 24
Horney, Mark 4, 214
Hunt, Nancy 153
Hutcher, Peter 280
Indictor, A. 329
Johnson, R. Dale 178
Johnson, Vivian 258
Kabasakalian, Rita 127
Kahn, Ted 255
Kalowaki, Phyllis 114
Karaffa, Jo Ann 173
Karlin, Marty 158
Kasten, Joyce 209
Kelberg, E. 329
Kimmich-Keyser, Andrea 127
King, Rebecca 257
Kmetz, Jack 48
Knezek, Don 224
Knezek, Gerald A. 278
Knight, Roger 24
Knupfer, Nancy Nelson 2
Kraver, Ted 318
Kressen, David 30
Kumar, M.S. Vijay 333
Labib, Adel 247, 270
Ladesou, C. 226
Lamb, Annette 289, 333
Land, Michael 158
Lange, Mary 80
Lathrop, Ann 335
Lawson, Virginia 247
Lawyer-Brook, Dianna 108
Lee, Mi Ok C. 295
Lehnen, Doug 287
Leung, Nelson K. 143
Levin, James A. 214
Levin, Sandra 90
Lewis, Jesse C. 250
Lieberman, Lynn 284
Lincoln, Bruce 26
Linsley, Barbara 108
Little, John 224
Lobeck, Sandy 335
Long, Madeleine 315
Lopez, Antonio 213
Lu, Cheng Dong 2
Lu, Mei-Yan 257
Lyness, Ann L. 217
Lyons, Rosemary 175
MacGregor, Kim 106
Maciejewski, Anthony A. 143
Macintosh, Patricia 333
Madeheim, Jim 241
Mahmoud, Emad 203
Marcus, Stephen 157
Markwell, Nancy 209
Martin, C. Dianne 203
Martinez, Carole 260
Matray, Kam 325
Matray, Paul 34
McCoy, Leah 258
McCullough, C. Sue 290
McGann, Sylvia 175
McGrath, Diane 171
McIsaac, Marina Stock 216
Meizel, Janet 280
Michela, Michael R. 278
Miller, Inabeth 28
Minichino, Camille 285
Moeller, Babette 262
Montoya, Ivy J. 204
Morris, Joyce 329
Mowe, Richard 49
Nelson, Christine 260
Nelson, Philip 90
Neufer, Eric K. 330
Newman, Denis 61
Nowak, Darlene 26
Noyce, Dorothy 226
Nydahl, Joel 257
O'Brien, Jim 345
Otto, Joseph C. 69
Overbaugh, Richard C. 7
Paige, Harriet 89
Papa, Frank J. 278
Papert, Seymour 249
Petrone, C. Tina 329
Pitre, Michel 138
Place, Ron 65
Planow, Mary 285
Poage, Julie 122
Poirot, James L. 140
Porter, Anne 48
Price, Bonnie 317
Raimond, Jean A. 217
Randlev, Karen 283
Reed, Michael 62
Reed, W. Michael 226
Reese, Paul 61
Rehfield-Pelles, Kathy 106
Reich, Carol 262
Repman, Judi 246
Rescigno, Ronald 330
Riel, Margaret 247, 270
Riley, Marlene 288
Rogers, M. 288
Rose, Richard 282

Rosen, Jane A. 156
Roth, Waldo 213
Rubenstein, Gayle 127
Rubin, Andee 63
Rucker, Rob 114
Ryder, Randall 110
Sandez, Beth 122
Sansing, Mike 65
Sarner, Ronald 285
Scali, Nancy 124
Scheiss, John 226
Scherer, Charlotte 76
Semrau, Penelope 74
Sesko, Stephen 285
Sgarlotti, Richard 78
Shores, J. 278
Short, Susan 31
Skinner, Kathleen 260
Smith, Janet 316
Smith, Jennifer 289
Smith, John M. 262
Smith, Linda 153
Smith, Richard Alan 110
Snelbecker, Glenn E. 132
Solomon, Gwen 125
Soloway, Elliot 62
Stager, Gary 91
Stapleton, Clinton E. 214
Steelman, Jane D. 141
Steinhaus, Kurt 5
Stemler, Luann 129
Stephens, Mary Margaret 116
Strange, John 74
Strange, John H. 158, 289
Strom, Robert 33
Strommen, Erik 149
Sullivan, Kathleen 173
Sutphen, Kathleen 116
Swafford, Bill E. 80
Swayne, Maria 33
Talaiver, Manorama 28, 173
Taylor, John T. 127
Tellep, Andrew 90
Tempel, Michael 256
Thompson, Ann 201
Thompson, Ann D. 295
Thornburg, David 276
Thrush, Emily A. 214
Tubb, Gary 118
Tyler, Doris 116
Van Dam, Janet 175
Ventura, Fred 277
Viau, Elizabeth 76
Vidville, C. 226
Vila, Joaquin 129
Wagner, Roger 32
Wall, James 74, 345
Walsh, Kevin 158
Waugh, Michael 204
Webb, Brian 285
Weiss, Marla 241
Weller, Herman G. 97
Weller, Martha H. 278
Wepner, Shelley 30
White, Charles 153
Wiebe, James H. 118
Williams, Brevard III 258, 280
Williams, Gerald 345
Williams, S. 329
Willis, Diana M. 141, 246
Wilson, Brent 138
Wilson, Judith D. 132
Wood, Billy 183
Wright, Kieth C. 54
Wright, William W., Jr. 201
Yoder, Sharon 25
Young, Jon I. 278
Zorich, Janelle 122

BEST COPY AVAILABLE